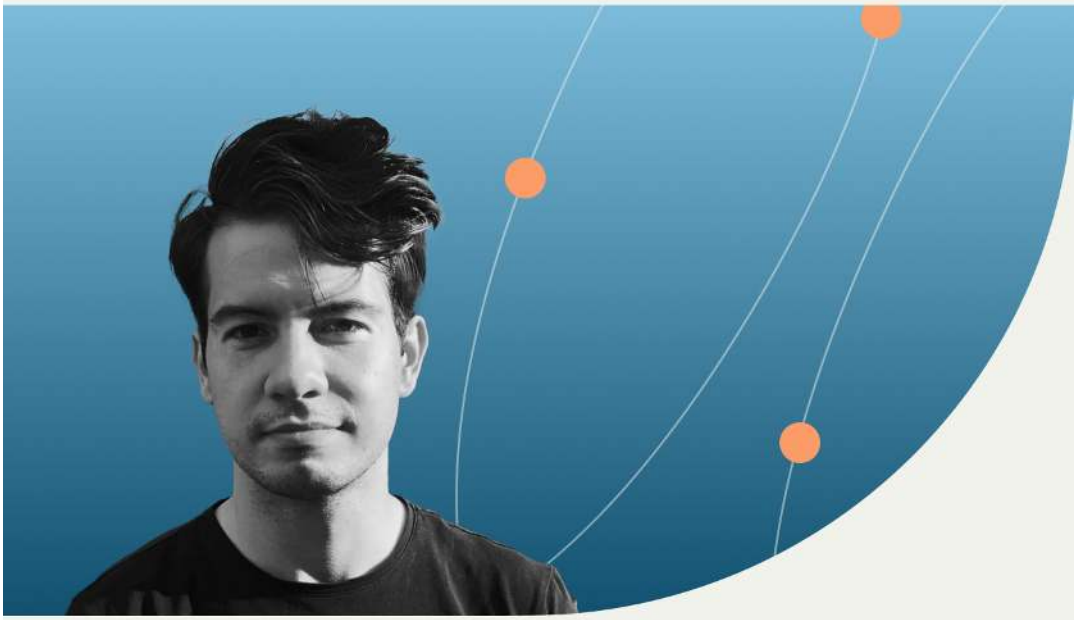
Tech Leader

ECHO Report

Generative AI for 80% Customer Support Automation

A spotlight on SOAX, a mid-sized global data extraction platform



Insights from **Kirill Markin**

Former Head of R&D at SOAX

O'REILLY

ECHO Report

Generative AI for 80% Customer Service Automation: A spotlight on SOAX, a mid-sized global data extraction platform

Insights from Kirill Markin

Former Head of R&D at SOAX

Contents

Landmarks

Acknowledgments

TechLeader Team

Managing Director: Ben Renow-Clarke

Portfolio Director: Manish Nainani

Head of Product: Levi Monaghan

Content Developer: Meera M. Bhagavathy

Program Manager: Ryan Shaw

Relationship Lead: Lucy Wan

Production Designer: Shankar Kalbhor and Ketan Kamble

Report Credits

Expert Insights: Kirill Markin

Chief Writer and Researcher: Meera M. Bhagavathy

TechLeader

Grosvenor House

11 St Paul's Square

Birmingham

B3 1RB, UK

ISBN 978-1-80611-278-4

techleader@packt.com

For all inquiries regarding this report or to discuss creating a report featuring your company's innovative work, please reach out to the TechLeader team at the above address.

About the Contributor

Kirill Markin

Former Head of R&D, SOAX

Current Director of R&D, Improvado

Kirill Markin brings over 12 years of experience in AI, data science, and business automation, with a proven ability to transform technical innovation into practical, impactful solutions. As a driving force behind SOAX's AI-powered customer support system, Kirill played a pivotal role in designing and implementing scalable architectures that balance efficiency, adaptability, and cost-effectiveness.

Kirill's technical expertise includes prompt optimization, pipeline automation, and developing modular AI workflows to streamline operations. His innovative use of task-specific models and real-time feedback mechanisms ensures SOAX's AI systems remain secure, agile, and aligned with business goals.

Describing himself as a "data monkey", Kirill thrives on experimenting with emerging technologies and continually pushing the boundaries of AI-driven solutions. His hands-on leadership has solidified SOAX's position as a leader in intelligent customer support automation.

Executive Summary

The global chatbot market is projected to grow at an impressive **CAGR of 24.32% by 2029**. This underscores a clear trend: AI is becoming the cornerstone of user interaction. As more and more enterprises adopt customer support automation as an essential workflow component, the real challenge will be in achieving intelligent automation that delivers meaningful value in the simplest and most cost-efficient way.

SOAX, a data extraction platform, has achieved 80% customer support automation through a multi-layered generative AI framework. Beyond efficiency, their AI assistant smartly escalates complex queries to human agents, striking a balance between automation and human expertise.

Discover how SOAX engineered layers to improve LLM reliability and accommodate crucial checks to identify, validate, and automate contextual cognition.

The problem

Like many global businesses, **SOAX** faced the challenge of providing 24/7 customer support while managing resource strain, repetitive queries, and scalability issues. With customers expecting instant, accurate responses, traditional human-only support models prove unsustainable.

The solution overview

- **An AI-powered customer support chatbot** that automates the easiest 80% of the repetitive queries and escalates the complex ones to human experts through a smart, contextual understanding of user-needs.
- **Development approach**: Phased, iterative, and agile, focusing on rapid functionality and a tight feedback loop.
- **Core technology integration**: OpenAI's ChatGPT, Go, Pinecone, Pipedream, Intercom, AWS, Docker.

Key outcomes and impact

- **80%** customer support automation
- **30-35%** reduced customer support hours
- **95%** rise in **customer satisfaction (CSAT)**

Key challenges addressed

- Data quality and documentation management
- Identifying the necessary layers and checks for engineering reliable responses and autonomous escalations
- Prompt engineering and optimization
- Integrating with existing workflows
- Balancing AI automation and human oversight

Key decisions

- Cross-referenced layers and chain-of-thought reasoning for contextual clarity and hallucination mitigation
- Organic integration of AI into the team's daily workflow
- Requirement-based use of LLMs
- Cross-functional collaboration

Table of Contents

How SOAX integrated its AI solution with existing workflows.

8. **Operational Challenges and Solutions**

The key challenges encountered by the team and how they addressed them, in problem-solution pairs.

9. **Key Insights for Tech Leaders**

Key findings of the case. Actionable insights for tech leaders looking to integrate a scalable and reliable customer chat-support system into their businesses.

10. **Conclusion**

Key insights with closing remarks.

11. **References**

Interface versus Intelligence: Rethinking Chatbots in Real-World Applications

The enthusiasm for AI-powered chat interfaces often emphasizes their potential for automation and cost reduction, sometimes at the expense of reliability and efficiency. Failures in real-world applications frequently stem from overlooked design, deployment, and management aspects. Key contributing factors include:

Inadequate training data: AI-driven chatbots depend on high-quality, diverse datasets to deliver accurate responses. Utilizing insufficient, outdated, or biased data can result in irrelevant or incorrect answers, leading to user frustration. For example, a chatbot trained solely on existing FAQs may struggle with nuanced or novel queries, failing to meet user expectations.

Lack of contextual understanding: Many AI chatbots lack the ability to grasp the context of conversations, particularly with complex, multi-turn interactions or ambiguous inputs. This limitation can cause misunderstandings, especially when industry-specific terminology or sequential problem-solving steps are involved.

Inability to handle escalations: Chatbots without defined escalation pathways often falter when encountering queries beyond their programmed capabilities, resulting in unresolved issues and negatively impacting user experience and brand perception.

Overreliance on static responses: Dependence on predefined responses or rigid workflows renders many chatbots incapable of adapting to dynamic or evolving user needs. This inflexibility hinders their ability to offer creative or personalized solutions.

Underestimation of edge cases: Developers may overlook edge cases, such as:

- Regional dialects.
- Informal language.
- Other rare issues.

These can lead to incomplete or unsatisfactory interactions. This oversight is particularly critical for global companies where linguistic and cultural variations influence communication.

Scalability challenges: As chat volumes increase, chatbots may struggle under heavy traffic due to inadequate backend infrastructure or poorly optimized response algorithms, causing inconsistent performance across different regions.

Neglected post-deployment monitoring: Continuous monitoring and retraining are essential yet often overlooked, leaving chatbots ill-equipped to handle evolving customer demands or new product features. Without a feedback loop for ongoing improvement, chatbots can become stagnant, diminishing their long-term value.

Overemphasis on interface over functionality: Focusing too much on sleek design and conversational ease can mask fundamental issues like low resolution rates or irrelevant responses. Prioritizing rapid deployment to match competitors may lead to insufficient testing and lack of iterative refinements.

Failure to align with business goals: In the rush to implement chatbots, organizations might adopt generic solutions that don't align with specific business objectives or customer expectations, resulting in a disconnect between anticipated benefits and actual performance.

Who is SOAX?

SOAX is a data extraction platform that delivers fast, reliable, and ethically sourced solutions to help businesses stay competitive and informed.

The company was founded in 2020 with a vision to revolutionize critical data extraction from the internet. Launched as a small-scale, residential proxy service, the company quickly recognized the growing demand for comprehensive data collection solutions across various industries and evolved into a full-fledged data extraction platform, catering to business-

es in e-commerce, marketing, media, and beyond.

Internal stakeholders

Co-founders: Stepan Solovev and Evgeny Kayumov.

Employees: Over 50 employees working remotely from across 15 countries.

The Pain-Point Drive to AI

Prior to the AI assistant, SOAX's customer support staff had to answer many simple questions repeatedly. Being a global company, these questions could arise at any time of the day, requiring staff to be on call around the clock. To alleviate this burden and allow customer support staff to focus on improving the company's documentation, SOAX decided to adopt an AI customer support assistant.

The main theory was, if we can move our human hours from all customer support departments and invest them instead on fine-tuning documentation, that would greatly enhance scalability."

--Kirill Markin

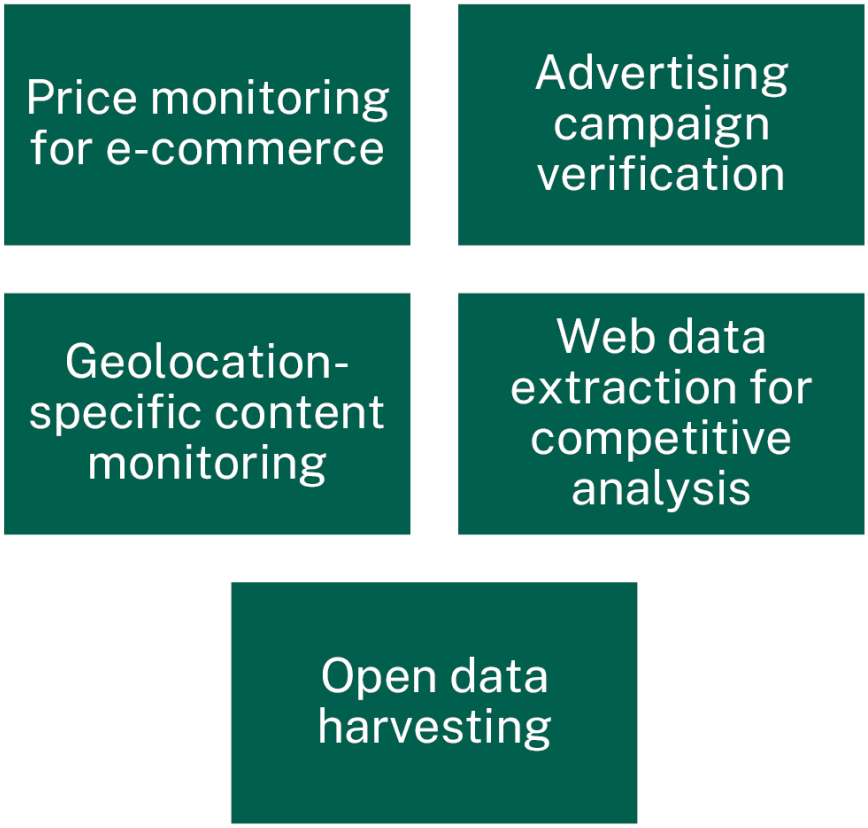


FIGURE 1: Key use cases SOAX takes care of.

The Solution: An AI-Powered Smart Chatbot

The smart chatbot solution combines a sophisticated RAG-based knowledge system with a multi-layered approach to generative AI. The layers incorporate various LLMs, sentiment analysis, proactive customer outreach, and a robust feedback loop.

Quantitative metrics

The initial emphasis was on operational efficiency and customer satisfaction, with cost optimization planned to be addressed in subsequent phases.

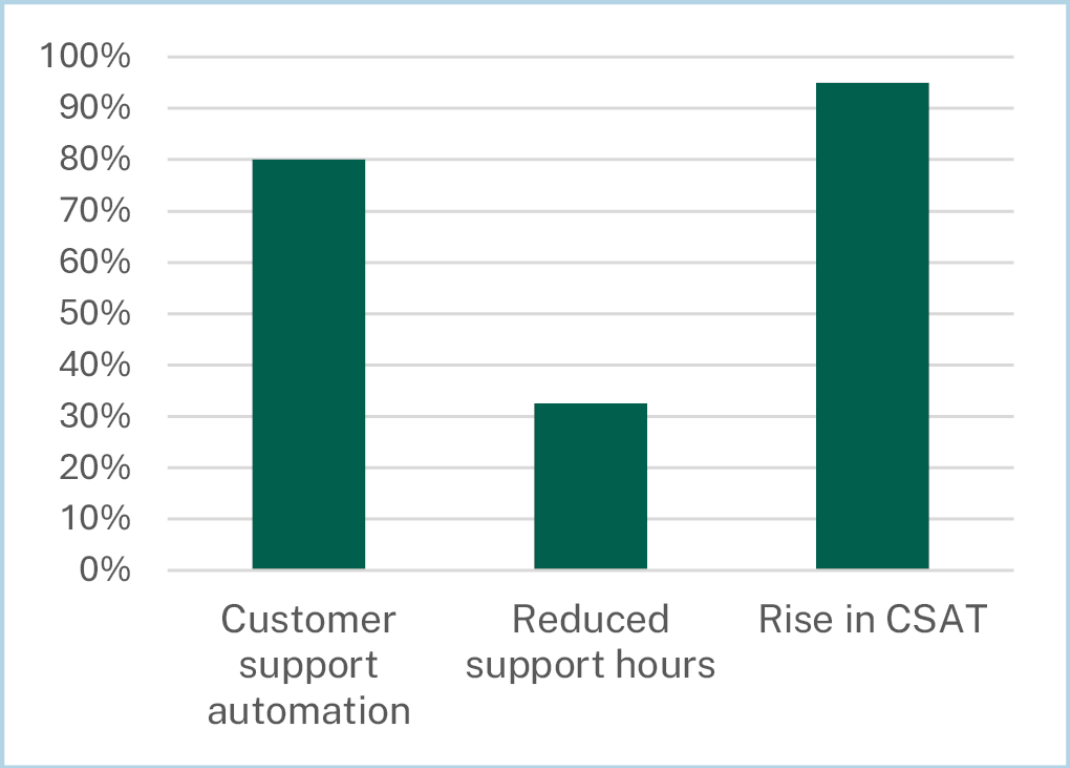


FIGURE 2: Quantitative outcomes of SOAX's chatbot solution.

Qualitative outcomes

Faster response times: The AI's near-instantaneous answers to common queries have significantly reduced customer response times, improving overall service efficiency.

Previously, our customers had to wait a day — sometimes even three — for support. Now, thanks to LLMs, they get answers in milliseconds, making resolutions faster without compromising accuracy."

-- Kirill Markin

Enhanced team efficiency: By handling routine inquiries, the AI has freed up the customer support team to focus on proactive initiatives, such as improving the quality, clarity, and structure of SOAX's documentation, enhancing both human- and AI-assisted interactions.

Proactive support capabilities: Integrated with SOAX's monitoring system, the AI enables proactive customer support by identifying real-time client issues and reaching out with potential solutions or resources, addressing problems before they escalate.

Unforeseen benefits

Skill development and growth: The project fostered internal career growth for team members, highlighting how AI initiatives create new opportunities for employees.

I must highlight Nikita, a former support technician who transitioned into an AI programmer during the development of this AI solution. He was the person responsible for the quality of answers and he got deeply interested in coding. Now, he creates AI systems like these."

-- Kirill Markin

Improved documentation: The AI project led to significant improvements in SOAX's internal documentation by uncovering and addressing inconsistencies, outdated information, and structural gaps, benefiting both internal teams and customers.

Near-human interaction quality: The AI's conversational fluency has reached a level where customers often cannot distinguish between chatbot and human interactions, underscoring its advanced design and effectiveness in customer satisfaction.

“

Many clients don’t realize they’re interacting with a bot. The AI adds irony, jokes, and smileys when the user wants to play with it and proactively sends URLs if no documentation is available within the database.”

-- Kirill Markin

Development and Deployment

Pre-launch decisions

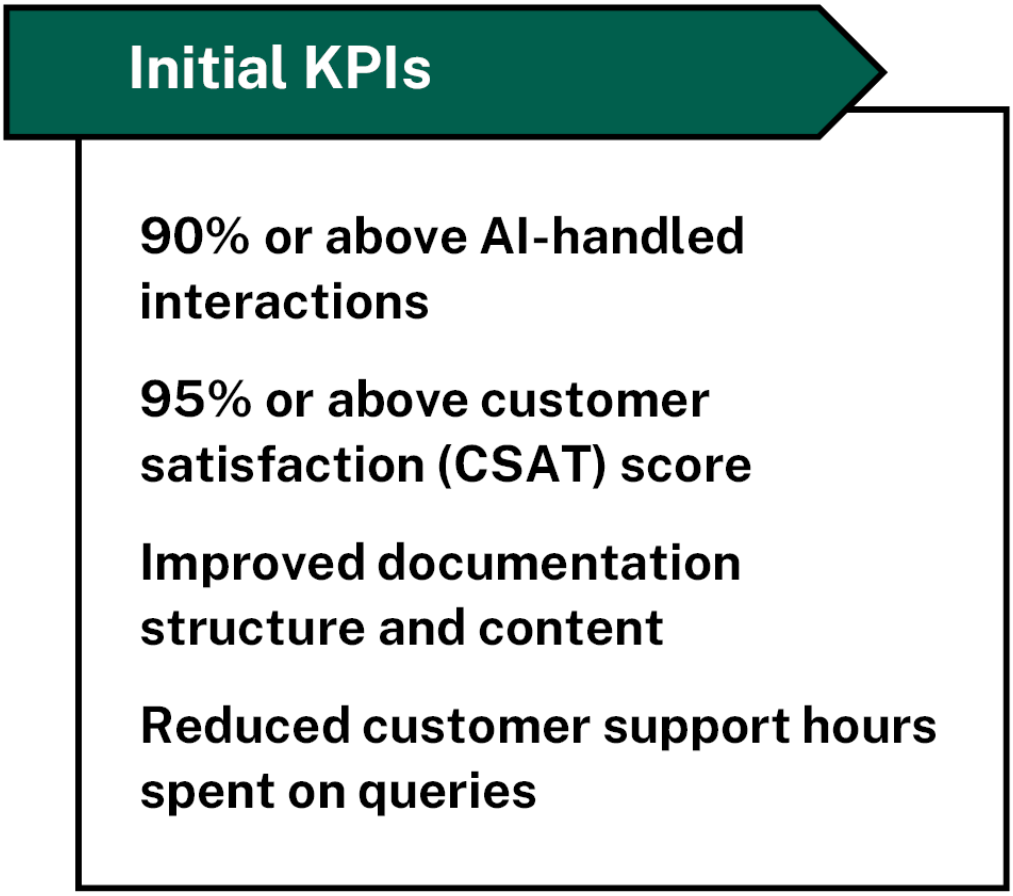


FIGURE 3: SOAX’s initial KPIs.

Core technology stack chosen for the AI solution

Data and storage

- **Pinecone**: A vector database for storing and querying embeddings generated from SOAX’s documentation, enabling efficient information

retrieval.

- **Pipedream Workflow builder**: For specialized tasks such as conversation quality analysis.

Cloud infrastructure

- **AWS (Amazon Web Services)**: Hosting solution for deploying microservices via Docker containers, aligning with industry-standard cloud deployment practices.

APIs and integrations

- **Intercom API**: Directly integrates the chatbot into SOAX's workspace, facilitating smooth interaction with customers through tasks like webhooks and message management.

- **Slack API**: Enables team members to add knowledge to the chatbot's database and engage with the AI for continuous improvements.
- **LangChain**: Orchestrates integrations with multiple LLMs and data sources for a seamless AI-powered workflow.

Machine learning and AI

- **LLMs:** Includes OpenAI's GPT models chosen based on task-specific requirements, cost, and performance.
- **RAG**: Combines LLM capabilities with efficient document retrieval to provide relevant responses.

Core programming and frameworks

- **Python:** Used for tasks involving AI integration, data processing, and workflow automation.
- **Go:** Implemented for the backend server to enhance performance, maintainability, and scalability. The Go service integrates with Pinecone and supports Slack and Intercom implementation.

After an initial week of MVP testing, the team made a deliberate transition from Pipedream to Go. The team achieved a simpler architecture and improved developer support with this decision.

Strategic transition from Pipedream to Go

This decision was driven by several key factors:

- **Long-term maintainability**: Go was selected because it simplifies development and ongoing support for the SOAX engineering team. Its efficiency and reliability make it a more sustainable choice for the production system.

- **Pipedream as a rapid prototyping tool**: The team used Pipedream as a temporary solution to quickly develop and test the MVP. It enabled rapid implementation and validation of service endpoints but was never intended as a long-term platform. The transition to Go was always part of the roadmap.
- **Strategic focus on functionality**: In the early stages, the primary objective was to validate core functionality rather than align with the existing tech stack. Pipedream, as a widely used tool, facilitated this process. However, with the MVP proving successful, rewriting the system in Go became the logical next step.
- **Optimizing for efficiency**: While Pipedream was useful for supportive processes such as conversation quality analysis, the core service required greater performance and scalability. Go provided the necessary robustness and efficiency to meet production demands.

The initial technology choice was not critical, as the team was prepared to rewrite the code entirely if the project proved successful."

-- Kirill Markin

SOAX's decision to deprioritize initial technology choices: A strategic overview

The SOAX team's decision to deprioritize the initial choice of technology reflects a strategic mindset driven by several key factors.

Speed of implementation: The primary goal was to rapidly test the feasibility of using AI for customer support. The team prioritized quick deployment over finding the "perfect" technology stack. This bias for action allowed them to prototype quickly and evaluate core functionality without delays.

"Plans are educated guesses," says Kirill, underscoring the importance of short-term, actionable goals that can adapt to new information and user feedback.

Learning and experimentation: The team viewed the project as a learning opportunity, particularly in experimenting with emerging technologies

such as LLMs and RAG systems. The willingness to experiment with various tools and gather user feedback early enables course corrections without significant sunk costs.

Resource efficiency: Aware of their limited resources, the team aimed to create a functional prototype with minimal initial investment. They selected tools pragmatically, knowing they could switch to a more robust solution later if their experiments succeeded.

Other key decisions

Leverage existing resources and expertise: Instead of hiring new staff, the team opted to utilize the existing customer support team's knowledge to train the AI and refine its responses. The team achieved this by prioritizing effective integration of the AI feedback pipeline with the existing customer support tools like Intercom and Slack. This is a cost-effective approach.

One of the problems with introducing AI tools into a testing workflow is that you can show them to your colleagues, see them play with it, but they will never open it again. The initial novelty doesn't transition into a sustained use. But if you implement it inside the interface, within their workflow, it will work."

-- Kirill Markin

Targeted focus on simple queries: Recognizing AI's limitations, the team focused on automating the most common and straightforward Q&A interactions. This approach enables human agents to focus on complex issues while AI efficiently handles the majority of routine customer interactions.

The build process

This section elaborates on how the AI chatbot was developed, starting from preparing the data to continuously testing the output. Four team members were involved, as well as four key departments:

- Customer care
- Sales
- Engineering
- R&D

FIGURE 4: The separation of responsibilities between the four-person development team involved in building SOAX's AI chatbot.

Data preparation using RAG

The team divided SOAX's existing documentation into smaller, interconnected pieces to facilitate retrieval by the RAG system, as small context windows result in better quality AI-generated answers. Keeping token size manageable ensures clarity and precision in responses, minimizing disruptions when processing large datasets. They used LangChain scripts to automate this process.

Later, they incorporated relevant information from marketing articles and Slack messages to improve the system's knowledge base.

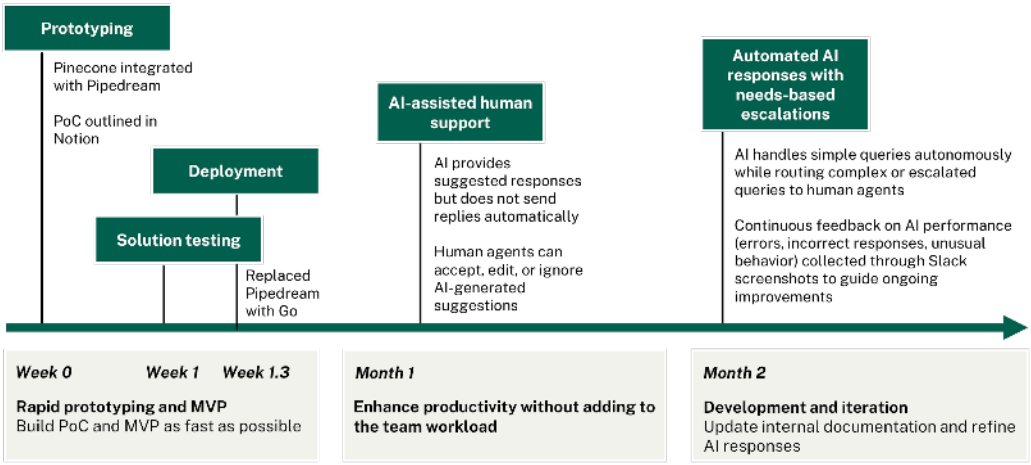


FIGURE 5: Development timeline of SOAX's AI-powered chatbot.

Updating the RAG and document database via Slack

SOAX employs a streamlined workflow integrating Slack with its document updating system. Team members can mark valuable Slack messages — internal conversation threads — by selecting the "Add to knowledge base" option. This sends the selected content to a webhook, which forwards it to a database for storage.

Daily sync: The system synchronizes daily, consolidating data from documentation, marketing materials, and Slack messages into the Pinecone database. While urgent updates can be requested as needed, the daily sync is typically sufficient.

Monthly manual review: Once a month, the Slack contributions are manually reviewed to assess whether they require updates to existing documentation or creation of new articles. A combination of automated daily updates with monthly manual reviews ensures that the RAG system and document database remain accurate, current, and relevant.

“

If you can build something in a week, why not give it a shot? Worst case, you lose a week of R&D — no big deal. The more you experiment, the higher the chances that some ideas will stick. In the early days, it might feel like everything is broken, and that’s totally fine. What matters is getting it out there. The system might be rough at first, but real users will test it, call it stupid, and that’s the best part — because now you know exactly what to fix.”

-- Kirill Markin

FEEDBACK TOUCHPOINT 1:

The structured, iterative approach to development illustrated in Figure 5 means that AI is gradually introduced, while ensuring quality and human oversight throughout the transition from manual to AI-assisted customer support.

SOAX’s AI-powered customer support architecture

This section outlines the architecture of the AI-powered customer support chatbot through five core workflow processes.

Customer interaction: Customers initiate conversations via Intercom, SOAX’s customer support platform. An Intercom API webhook triggers the AI chatbot when a new conversation starts.

AI chatbot processing: A Go-based server receives conversation details from Intercom. The server interacts with a LangChain script to prepare data for processing by LLMs.

The LLM processes conversations using prompt engineering and chain-of-thought techniques to understand user intent and generate responses.

Information retrieval: The LLM integrates RAG to fetch relevant information from SOAX's documentation stored in Pinecone. Pinecone returns documentation snippets based on the semantic similarity of the user query.

Response generation and decision-making: The LLM combines retrieved documentation with the conversation context to craft responses. Additional layers ensure security, sentiment analysis, and escalation decisions.

Response delivery and feedback: If escalation is not needed, the AI sends the response directly to the customer via Intercom. If human intervention is necessary, the conversation is seamlessly handed off to a human agent using Intercom's built-in functionality.

The customer support team uses a **custom Chrome extension** to review AI-generated responses and provide feedback. They add insights and updates to the documentation through Slack, enhancing future AI interactions. A continuous feedback loop ensures the AI improves over time, refining accuracy and effectiveness based on real-world interactions.

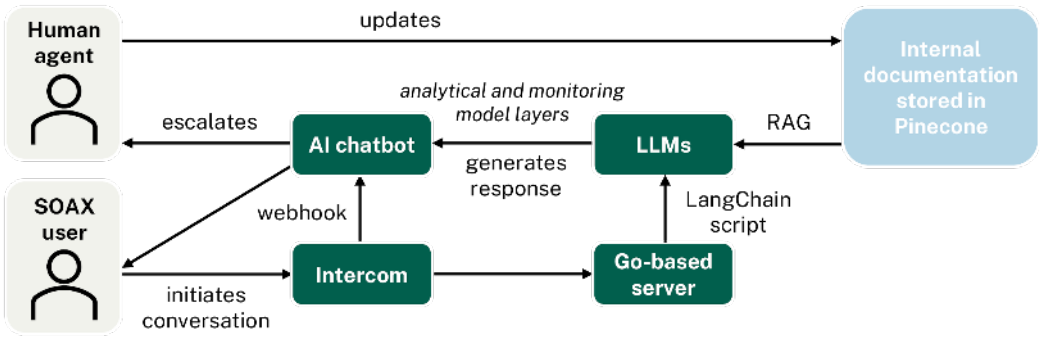


FIGURE 6: The AI chatbot architecture.

The AI-human switch

This section introduces the processes governing the transition from AI-driven interactions to human intervention. It explores the specific conditions under which chatbot interactions qualify for escalation, describes the underlying escalation mechanisms, and outlines the multi-layered es-

calation pathways explicitly designed to address and mitigate potential inaccuracies inherent in LLMs.

Conditions for escalation

SOAX's AI-powered customer support system handles most inquiries, but includes mechanisms for escalation to human agents under specific conditions. Below are the 4 conditions of escalation.

Condition #1: User requests human interaction. The AI detects explicit user requests for a human agent using keyword analysis through a non-LLM AI model.

Condition #2: Complex or sensitive topics. The AI relies on its knowledge base (documentation, marketing articles, and curated Slack messages) but escalates when:

- Payment-related issues arise.
- Legal or compliance inquiries are made.
- Technical issues exceed the documentation's scope.

When faced with queries outside its knowledge base, the AI offers informed guesses while clarifying its limitations. If the user is unsatisfied, escalation is triggered.

Condition #3: Implicit user dissatisfaction. The AI monitors for frustration or dissatisfaction, even without explicit human-agent requests. Sentiment analysis identifies patterns of:

- Repeated failed attempts to resolve issues.
- Users circling the same topic without resolution.

Two key features of the AI chatbot architecture

- **Simple yet effective**: SOAX developed agent-like behaviors for their chatbot through a hard-coded pipeline with multiple LLM calls. This emphasizes simplicity and maintainability.
- **Leverages existing documentation**: The system relies heavily on well-structured documentation, highlighting its importance in achieving high-quality responses.

The AI escalates after identifying consistent dissatisfaction, ensuring a measured response instead of reacting to isolated incidents.

Condition #4: Security and inappropriate content. The system prevents engagement in discussions involving illegal or inappropriate topics (for example, drugs). In such cases, the AI ceases interaction and escalates to the security team for further review.

Escalation mechanism

A dedicated decision-making layer analyzes conversation flow, sentiment, and topic complexity using rules and curated examples informed by customer support input. The system adapts dynamically, incorporating new rules and examples based on agent feedback and conversation analysis.

Here's how this dynamic adaptation works: The chatbot's escalation mechanism dynamically adapts in real time based on continuous inputs from customer support feedback and documentation updates. Initially, the AI may escalate interactions too early or miss opportunities to escalate when necessary. By analyzing direct feedback from the customer support team, the chatbot continuously refines its escalation logic. Even if the initial messages do not indicate complexity clearly, the system reassesses each subsequent message and customer responses, dynamically correcting earlier judgments. Additionally, ongoing updates to internal documentation help the chatbot stay current, enabling consistently accurate, contextually relevant decisions aligned with evolving products, policies, and procedures.

Layered architecture for resolving LLM fallibility

SOAX's AI system employs a modular, layered architecture to enhance accuracy, reliability, and flexibility. Each layer serves a specific function, addressing LLM overconfidence and hallucinations effectively. The following is a breakdown of the layers involved in the escalation pathway.

1. **Core LLM pipeline**: At the heart of the system is the core LLM pipeline, covered in the previous section.
2. **Sentiment analysis**: Evaluates user messages to detect emotional states. For instance, if frustration is identified, the AI adjusts its tone or escalates the conversation to a human agent promptly.
3. **Conversation flow monitoring**: Tracks user-AI interactions to identify repetitive failures or lack of resolution. If the conversation stalls, the system triggers escalation to a human agent.
4. **Security and compliance checks**: Ensures adherence to SOAX's policies by preventing the AI from disclosing sensitive information or engaging with inappropriate topics.
5. **Optimizing prompting and chain-of-thought reasoning**: SOAX refines

prompts to guide the LLM effectively and uses chain-of-thought (CoT) reasoning to break down complex problems into smaller, context-specific steps. Tasks are split into calls, such as asking clarifying questions (call 1) and providing answers (call 2). This iterative approach improves contextualization, reduces hallucinations, and enhances multi-step problem-solving accuracy.

6. **Task decomposition and multi-model verification**: Complex tasks are divided into smaller sub-tasks, enabling the use of task-specific LLMs instead of over-relying on large, expensive models. Responses are verified through multiple models, ensuring accuracy and avoiding unnecessary clarifications. This decomposition prevents overconfidence and optimizes resource usage.

7. **Grounding and escalation rules**: Explicit escalation rules reduce the risk of the AI providing incorrect or incomplete responses. The RAG system ensures that responses are grounded in SOAX's documentation, aligning outputs with the company's knowledge base and products.
8. **Human-in-the-loop feedback**: Customer support agents flag problematic responses, provide context, and update the knowledge base, forming a critical feedback loop. This process mitigates LLM overconfidence and hallucinations, driving continuous system improvement.
9. **Escalation and human handover**: When escalation is necessary, the AI uses Intercom's "Switch to colleague" function to notify both the human agent and the customer, ensuring a smooth handover. The system prioritizes speed in escalation, reducing reliance on perfect accuracy and minimizing the impact of incorrect assessments.

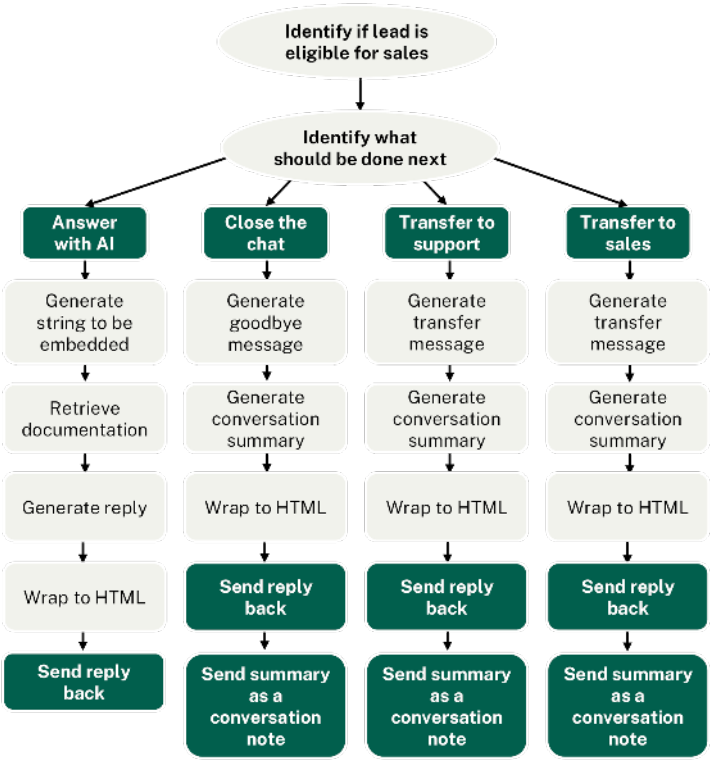


FIGURE 7: SOAX's model layers diagram.

Ethical and Security Compliance

Data sanitization techniques

Removing sensitive user data: At the core of SOAX's data privacy strategy is the proactive removal of personally identifiable information (PII) and other sensitive details before data is sent to any LLM. This foundational

step safeguards privacy while maintaining compliance with regulatory standards.

"You start privacy by removing the user-sensitive data from other ends, such as from LLM calls," explains Kirill, underscoring the importance of sanitizing data at the outset.

Transitioning to custom models: In the long term, SOAX aims to develop and host its own custom AI models. Kirill views this as the most secure option, providing full control over data processing and eliminating dependency on third-party systems.

Key decisions

Prioritize publicly available data

SOAX's customer support interactions primarily rely on publicly accessible information or documentation. This approach minimizes the risk of exposing private or sensitive data. However, the team remains vigilant, ensuring that even publicly available data is sanitized before being sent to LLMs for processing.

Leverage open-source libraries to balance security and business goals

Rather than building custom data sanitization libraries from scratch — a process Kirill describes as complex and resource-intensive — SOAX relies on widely used open-source libraries available on platforms like GitHub. These proven solutions streamline the sanitization process while allowing the team to focus on their core business goals.

"Don't try to create this library on your own," Kirill advises, emphasizing practicality in leveraging existing tools. "You should prioritize the core business." He advocates for efficiency and alignment between security measures and the company's primary business objectives.

Security and compliance

Security testing and red teaming: SOAX employs an internal red team strategy for proactive security testing. Programmer colleagues are invited to attempt to "break" the system, identifying vulnerabilities before deployment. Additionally, the dedicated security team collaborates during AI development to ensure adherence to SOAX's security protocols and industry best practices.

GDPR and legal compliance: SOAX consults its legal team to ensure the AI system complies with GDPR and other relevant regulations. The legal team identifies ethical and legal risks, advising on mitigation strategies to ensure compliance and minimize liabilities.

Transparent handoff to human agents: SOAX is transparent with customers about AI interactions. When escalation to a human agent is required, the system uses Intercom's built-in agent-to-agent transfer features, ensuring an efficient handoff process without requiring custom code.

In the future, I imagine disclosure statements on AI involvement becoming redundant. There won't be statements like, 'This program uses AI.' Of course it does!"

-- Kirill Markin

Implementation and Integration

SOAX implemented its AI customer service system through a structured, multi-stage rollout. The process began with a successful 3-week MVP test and evolved into a fully integrated AI assistant supported by a Chrome extension and continuous feedback mechanisms. This approach minimized disruption while maximizing learning and iterative improvement.

Integration and feedback process

1. Intercom API integration

- SOAX utilized Intercom's API to create webhooks for specific events, such as "Conversation created".
- A new app was added to the Intercom UI, and a webhook was created to trigger the AI bot for new conversations. Bot replies were sent back to Intercom via the API.

2. Closed-loop feedback for continuous improvement

- The AI pulls responses from a predefined knowledge base containing documentation, FAQs, and company guidelines.
- When a customer escalates or reports inadequate responses, customer support agents review the conversation and assess the AI's knowledge source to determine if:
  - Documentation is outdated or incorrect.
  - The AI misinterpreted information.
  - The context of the query was misunderstood.

Agents directly update documentation to:

- Clarify ambiguous language.
- Add details to cover edge cases.
- Correct outdated information.

Real-time updates to the knowledge base ensure immediate alignment with accurate information, accelerating improvements and empowering agents to take ownership of the AI's performance. The team tracks the impact of these updates on AI performance and customer satisfaction.

3. Chrome extension and human-in-the-loop recommendation phase

- A Chrome extension was developed to integrate suggested AI-generated responses into Intercom.
- Human agents reviewed and edited suggested responses for accuracy, tone, and context before sending them to customers.
- Feedback was provided to the development team via screenshots of incorrect or inadequate responses, enabling iterative daily refinements.

4. Role of LangChain and manual revisions

LangChain automated technical aspects of documentation division, but manual revisions remained essential. The Customer Support team identified inaccuracies or outdated information, rewriting it to ensure clarity and accuracy for both humans and AI.

5. Automated responses and dual integration

- After achieving sufficient accuracy, the AI assistant sends automated responses via Intercom.
- For unresolvable queries, the AI uses Intercom's "Switch to colleague" feature to escalate to human agents.
- The AI is also integrated into the team's Slack workspace, particularly in the "Questions and answers" channel, to streamline workflows without introducing unnecessary tools.

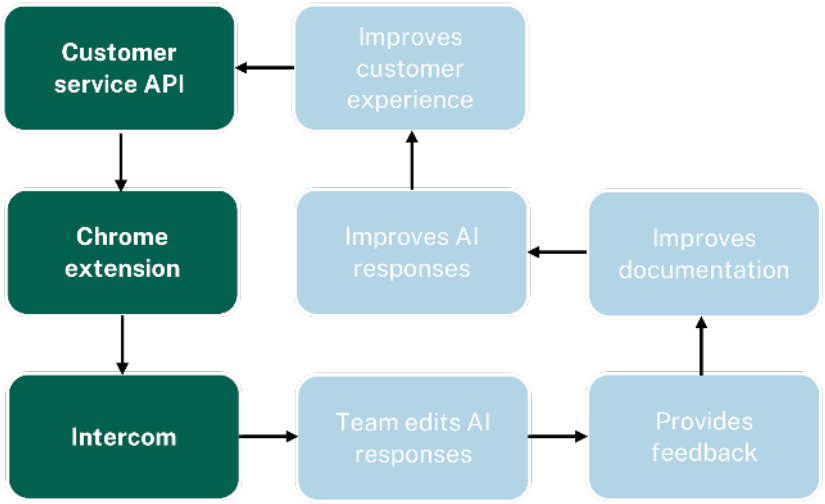


FIGURE 8: The closed feedback loop workflow.

Operational Challenges and Solutions

While SOAX's agile approach and emphasis on speed minimized some obstacles related to project management, development, and operational efficiency, several key challenges emerged.

Challenges faced by the customer support team

Shift in workflow: The implementation of the AI assistant required a significant shift in the workflow of the customer support team. Agents transitioned from primarily answering customer queries to focusing on documentation improvement and handling escalated cases. This change required adaptation and training for the team. The shift was marked by the following requirements:

- **Documentation clarity and data sensitivity**: The customer service team needed to ensure the documentation used to train the AI was clear, concise, and free of sensitive data. This involved a thorough review and potential rewriting of existing documentation to meet these criteria, and adding security prompts to prevent jailbreaking.
- **Ongoing prompt optimization**: A key challenge was the continuous optimization of prompts to reduce the number of incorrect decisions made by the AI regarding how to handle conversations. This involved analysing customer interactions, identifying areas where the AI struggled, and refining the prompts to improve the chatbot's decision-making accuracy.

- **Initial high volume of feedback**: During the testing phase, the customer support team was inundated with screenshots and reports of the AI's performance, requiring significant effort to analyze and address the feedback.

Solution: Effective integration of the AI into the daily customer support staff workflow

- A custom chrome extension for feedback, as explained in Section 7
- Slack and Intercom integration for knowledge updates and team adoption
- Continuous feedback loop for regular documentation updates

Outcome: A significant reduction in agent workload and improvement in the quality of customer interactions.

The key to getting the best out of new tools isn't just introducing them — it's organically integrating them into existing workflows. You don't want to pile extra work on customer support teams; you want to take work off their plate. The best way to do that? Meet them where they already work. Embed the tool right into their interface, make it intuitive, and let it naturally become part of their process. That's how you drive real adoption and results."

--Kirill Markin

Additional operational challenges

Challenge 1: Identifying the necessary layers of analysis

Determining the optimal number and complexity of analytical layers for the AI system proved to be a significant challenge. Finding the right balance between adding extra checks for security, tone of voice, and accuracy, while avoiding unnecessary complexity and cost, was a constant balancing act. This required careful consideration of potential risks and benefits, relying heavily on feedback from the customer support team, and real-world testing to guide their decisions.

Solution: Smart stacking with targeted self-analysis

SOAX tackled this challenge with a smart stacking method, incrementally adding layers of oversight and refinement based on targeted feedback, real-world testing, and continuous analysis. Their approach prioritized identifying weak points, optimizing data quality, and refining AI responses through systematic self-attention layers.

Iterative and cost-effective layering for precision: Fail fast, learn faster

SOAX started with a minimal implementation and stacked additional layers of analysis over time. Rather than over-engineering from the outset, they used a “fail fast, learn faster” mindset, allowing them to pinpoint flaws early and introduce corrective mechanisms efficiently.

They ranked potential improvements based on impact, cost, and efficiency, and discovered that a blend of simple and sophisticated methods often outperformed standalone complex solutions. This balanced approach allowed them to stack layers strategically without excessive overhead, making their system both robust and scalable.

Targeted self-analysis through feedback loops

A key component of their approach was real-time feedback from human operators. The customer support team actively flagged incorrect, misleading, or incomplete AI responses, helping engineers identify necessary self-attention layers. Additionally, SOAX employed **red teaming**, a process where programmers deliberately attempted to break the AI and provoke flawed outputs. This stress-testing approach revealed systemic weaknesses and guided adjustments to prevent recurring errors.

Data quality as the first layer of defense

SOAX recognized that AI performance is only as good as the data it learns from. In many cases, poor responses stemmed not from the AI itself but from incomplete or ambiguous internal documentation. By refining and expanding their knowledge base, they ensured the AI had clearer, more reliable sources to pull from, leading to improved accuracy without requiring additional technical complexity.

Monitoring and behavioral adjustments

To maintain ongoing improvements, SOAX implemented a feedback pipeline for conversational analysis to track the AI's behavior. It allowed them to fine-tune tone, response quality, and escalation triggers, ensuring that the AI adapted dynamically to user needs. This is explained in detail under the solution for Challenge 2.

Outcome: SOAX created a replicable model for refining AI reliability through a structured feedback approach. The system became adaptable, efficient, and responsive to real-world challenges. This approach proves that structured iteration, rather than a one-time fix, leads to long-term AI success.

Challenge 2: Managing the overconfidence and hallucinations of LLMs

SOAX encountered the common problem of LLMs being overly confident in their responses, even when lacking sufficient information to provide accurate answers. This tendency to prioritize user satisfaction over accuracy posed a significant challenge. Besides the overconfidence, LLMs are also mathematically proven to hallucinate — the tendency to generate incorrect or nonsensical responses.

Solution: Guided four-step reasoning – Stop, slow down, clarify, and cross-validate

SOAX implemented strategies like breaking down complex tasks into multiple LLM calls with more specific instructions, forcing the AI to slow down and request clarification when needed.

Prompt chunking/decomposition

The team breaks down complex tasks into multiple steps, or "two different LLM calls", and provides specific instructions for each call:

- The first call focuses on a specific aspect of the task, such as asking the user additional clarifying questions.
- Only after this initial call is completed does the LLM proceed to the second call, which focuses on generating the final answer.

This separation of tasks allows the LLM to focus on one instruction at a time, potentially reducing overconfidence and improving accuracy. Overall, prompt chunking allowed for more control over the AI's responses and encouraged it to ask clarifying questions when needed.

Adding cross-reference layers: SOAX uses an Intercom feedback loop to analyze customer interactions. This internal system has a structured pipeline for conversational analysis, with the core functions executed through a non-LLM based natural language processing (NLP) approach. LLM-based agentic systems monitor the bot's behavior regularly.

Below is a breakdown of the non-LLM based pipeline:

1. **Topic and sub-topic analysis**: Identifies recurring themes, highlighting chatbot strengths and areas for improvement.
2. **Customer segmentation**: Uses customer details to tailor insights for specific user types or demographics.
3. **Chat ID tracking**: Assigns a unique identifier to each conversation, allowing targeted retrieval and detailed analysis.

4. **Sentiment analysis and intent classification**: Assesses user sentiment and categorizes intent to refine the chatbot's responses.
5. **Prompt optimization**: Evaluates the effectiveness of prompts and iterates them to enhance the chatbot's performance.
6. **Documentation enhancement**: Identifies gaps in the knowledge base and updates documentation to address frequently asked questions or misunderstood topics.

Functions 1 to 4 can be performed by traditional ML models and NLP techniques. Functions 5 and 6 are accomplished by the customer support team. Function 5 involves A/B testing with human-in-the -loop evaluations and Function 6 can be executed through a combination of query logs and manual review updates.

The LLM-based agents analyze the bot's performance for tone, trends, and derive actionable insights for improvement.

Key decision: Use a combination of LLMs, non-LLM based models, and human-in-the loop evaluations to reduce the number of calls to the LLM, control the cost, and optimize performance.

Outcome: Cross-validation layers provided opportunities to add functions such as irony, jokes, or links to the documentation. These layers also forced the LLM to consider alternative perspectives. The AI system demonstrated more accuracy and dynamic adjustments to user needs.

For me, it's just one model doing a bunch of different things — just a couple of files, a few lines of code. Instead of burning through token limits answering every single question, we use keyword routing to point users to the right docs whenever we can. It's way cheaper, way faster, and honestly just makes more sense."

-- Kirill Markin

Factor	Non-LLM models (ML/NLP)	LLM-based AI agents
Reliability and predictability	High (rule-based, deterministic)	Medium (generative, contextual but can hallucinate)
Processing speed and cost	Low computational cost	High inference cost
Interpretability	Transparent and explainable	Opaque (black-box nature)
Best use cases	Structured analysis, classification, data tracking	Contextual analysis, insights extraction, tone adaptation

TABLE 1: Non-LLM models versus LLM-based AI agents by factors like reliability, cost, and interpretability.

Challenge 3: Striking a balance between automation and human oversight

While automation was a primary goal, SOAX recognized the importance of maintaining human oversight and intervention capabilities within the AI system. Determining the appropriate level of human involvement, particularly in handling complex or sensitive queries, was crucial. This required a deep understanding of the AI's capabilities and limitations and a well-defined escalation process to ensure a seamless transition from AI to human support.

Solution: 80-20 balance

SOAX aimed to automate the "easiest 80%" of customer interactions, leaving the remaining 20%, often requiring human judgment and expertise, to human agents. This was achieved by training the model to recognize complex queries or user frustration and having it automatically bring a

human operator into the loop to address the query. This ensured zero drop in quality while significantly scaling the operation.

The hardest part of creating the solution is not the programming; it is generating the ideal prompt exactly tailored to your business. In this prompt and example creation, the main people involved are the subject matter experts, because they understand why you should do one thing here and why you shouldn't do the same thing there."

--Kirill Markin

FEEDBACK TOUCHPOINT 2:

Key Insights for Tech Leaders

Multilayered generative AI framework for reliability and contextual clarity

A multilayered architecture overcomes key limitations of LLMs, like overconfidence and hallucinations, to ensure robust and efficient automated customer support.

On the foundational layer, retrieval-augmented generation integrates internal documentation with custom-tailored prompts to form the core schema for response generation and proactive customer engagement. Complementing this, the "pause and reflect" mechanism, driven by layered logic, ensures contextual validation, and mitigates the risk of inaccurate or irrelevant outputs. Together, these components create a scalable, intelligent architecture that balances automation efficiency with response reliability.

Model-agnostic architecture

Kirill emphasizes that the assistant is not tied to any specific LLM and can readily incorporate new models as they become available. This adaptable

approach future-proofs their system and allows for continuous improvement by integrating cutting-edge LLMs.

Rigorous cross-functional collaboration

The team recognized that high-quality documentation was essential for training the AI assistant. The customer support team transitioned to improving and expanding existing documentation, identifying inaccuracies or gaps highlighted through AI interactions. Collaboration between the R&D, customer support, security, and legal teams was crucial to overcoming challenges.

Enable autonomy for non-technical staff in the AI-improvement pipeline

Enabling non-technical staff to manage AI documentation accelerates enhancements in response quality. Non-technical staff who are well-versed in what the end user requires and appreciates are a cornerstone in the AI system improvement workflow. Actively involving them in updating, editing, and revising documentation optimizes resource efficiency, enables teams to discover diverse insights, and delivers real-time feedback. It leads to improvements that align more closely with user needs and expectations. This autonomy not only accelerates the enhancement process but also cultivates a culture of ownership and collaboration, enhancing the effectiveness of AI systems.

A strict requirements-based use of LLMs

The DevOps team at SOAX realized that LLMs should be treated as tools rather than goals. A judicious separation of functions that require LLMs and those that do not ensures cost-effectiveness, smoother pipeline designs, and faster deployment.

For structured tasks (categorization, tracking, intent recognition), non-LLM models are more stable, cost-effective, and reliable. For nuanced insights (tone analysis, summarization), LLMs excel but require oversight to mitigate errors.

A hybrid approach maximizes efficiency, combining LLMs for language understanding and non-LLM techniques for structured accuracy and cost control. This strategic balance optimizes chatbot reliability and intelligent

adaptability.

Balancing speed and responsibility

SOAX's AI project highlights the value of experimentation and rapid prototyping, emphasizing their role in its success. The success of the AI customer support project, stemming from a one-week MVP, reinforces this approach and encourages a more agile and innovative mindset within the company.

Don't worry too much about quality during the initial MVP stage. I'd prioritize moving faster, even if it involves some risks. The speed of delivery is more crucial than we often realise. Speed is closely tied to the resources we invest."

--Kirill Markin

Future optimization strategies

Optimizing the cost of using LLMs without compromising response quality is a top priority for SOAX. Currently, its pipeline requires **fewer than 10 LLM calls per interaction**. To further streamline costs, SOAX plans to do the following things.

Streamline task allocation: Delegate simpler tasks, such as formatting responses for Intercom or detecting irony, to less expensive models or rule-based systems. By reserving powerful LLMs for complex tasks, this strategy optimizes resource usage while maintaining high response quality.

This approach reduces API calls to LLMs, significantly lowering costs while preserving the effectiveness of their AI-powered customer support.

Power business and the tech core with AI: The future of enterprise AI and SOAX: Kirill envisions AI playing an increasingly pervasive role in all aspects of SOAX's business operations. He sees particular potential in applying AI to core technical processes and in enhancing their core product offerings, such as code generation and optimization and providing data-driven insights for decision-making. AI-powered business and tech core is a

strong trend that he foresees as the ideal future for enterprise AI.

FEEDBACK TOUCHPOINT 3:

Conclusion

The SOAX case study offers tech leaders a roadmap for leveraging generative AI to build an AI assistant significantly more intelligent and capable than a standard chatbot. By deploying a multi-layered framework coupled with a dynamic model selection, SOAX tackled common LLM challenges while achieving 80% automation. The strategic combination of RAG-based information retrieval, tailored prompt engineering, and chain-of-thought reasoning ensured accurate and context-aware responses.

Key to SOAX's success was an agile, iterative development process that favored rapid MVP delivery and continuous improvement over rigid, long-term planning. This approach allowed the team to remain flexible, quickly address bottlenecks, and refine their system in real-time. Seamless integration with Intercom and Slack ensured smooth workflows, while sentiment analysis and escalation protocols bridged the gap between AI and human intervention.

For tech leaders, this case emphasizes that achieving impactful AI adoption requires blending technical innovation with cross-functional collaboration and aligning AI capabilities with business needs.

Closing thoughts

Chatbots and AI-powered customer support assistants have become integral across various industries, enhancing efficiency and customer engagement. Initial projections underscored that by 2025, 95% of customer interactions will be AI augmented. AI-powered chatbots are no longer just support tools; they are emerging as strategic drivers of customer engagement and sales. Beyond automating responses, they can actively enhance user experience, build brand loyalty, contribute to increased customer satisfaction, and directly impact business revenue through intelligent engagement.

Enterprises that are looking to achieve the dual benefits of integrating AI into customer-facing roles cannot overlook SOAX's layered self-attention design and proven strategies for executing it efficiently.

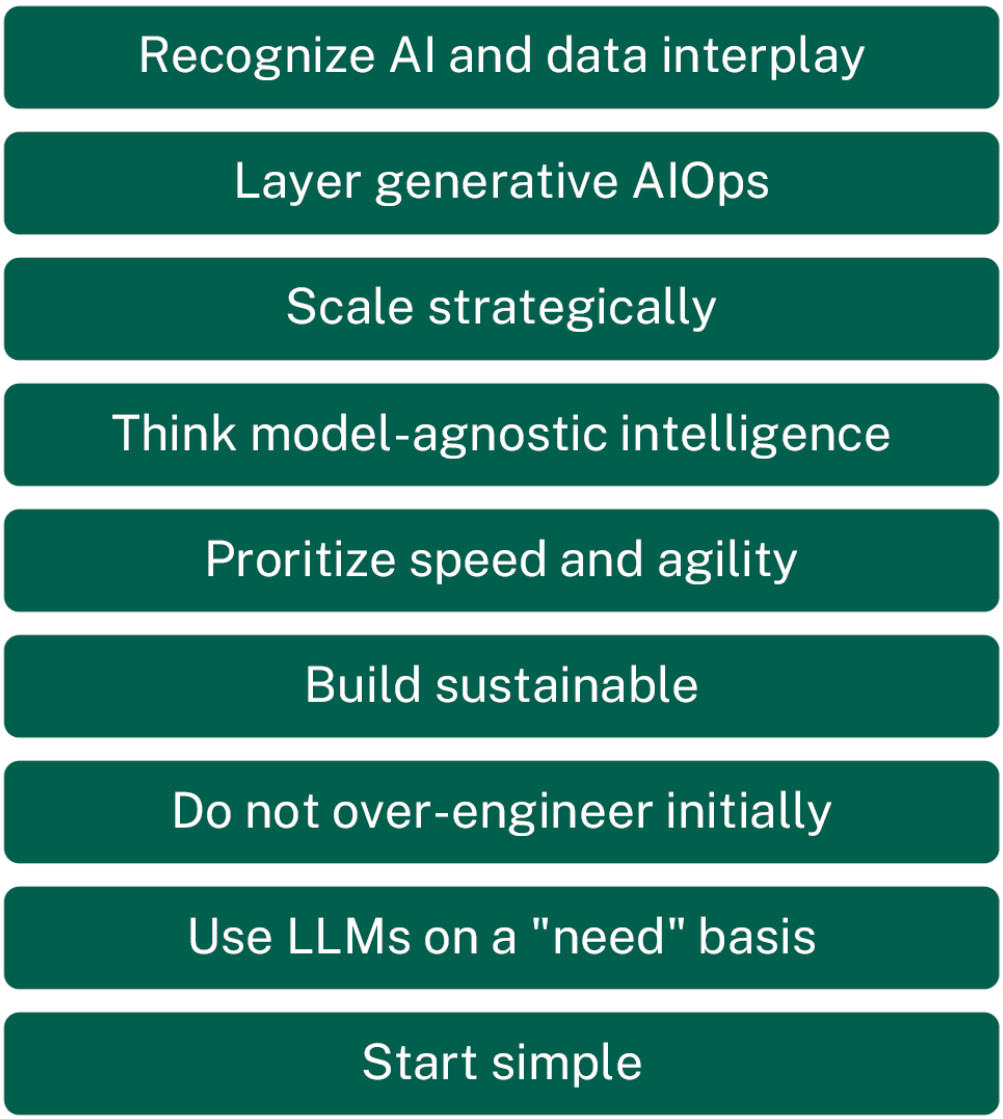


FIGURE 9: Key takeaways for tech leaders.

Next steps for the Strategist

- **TELL US**: Where do you draw the line between LLM and non-LLM in your architecture, and how is this evolving?
- **TELL THE CONTRIBUTOR**: Where do you disagree with Kirill the most?
- **ANSWER FOR YOURSELF**: Which teams would benefit the most and least from a generative AI transformation at your organization?

Next steps for the Visionary

- **TELL US**: How do we future-proof against technical drift and organizational misalignment at the same time?
- **TELL THE CONTRIBUTOR**: What problems could Kirill help you with?
- **ANSWER FOR YOURSELF**: What is the most high-risk, high-reward bet you could take at your organization at the moment?

References

- **Amazon Science (n.d.) How task decomposition and smaller LLMs can make AI more affordable. Available at**: Amazon Science (Accessed: 1 February 2025).
- **Banerjee, S., Agarwal, A., and Singla, S. (2024) LLMs Will Always Hallucinate, and We Need to Live With This. arXiv preprint arXiv**:2409.05746. Available at: https://arxiv.org/pdf/2409.05746 (Accessed 1 February 2025).
- **Centre for Security and Emerging Technology (2024) What does AI red-teaming mean? Available at**: CSET (Accessed: 1 February 2025).
- **Chokkattu, J. (2025) 'Your Pizza Guy Is Now AI', WIRED, 30 January. Available at**: https://www.wired.com/story/palona-ai-chatbot-salesperson/ (Accessed: 3 February 2025).
- **Dsouza, A., Glaze, C., Shin, C., and Sala, F. (2024) Evaluating Language Model Context Windows**: A "Working Memory" Test and Inference-time Correction. arXiv preprint arXiv:2407.03651v2. Available at: https://arxiv.org/pdf/2407.03651v2
- **Gartner (n.d.) Customer service artificial intelligence. Available at**: Gartner (Accessed: 1 February 2025).

- **Google Cloud (n.d.) Create a self-escalating chatbot in conversational agents using Webhook and Generators. Available at**: Google Cloud (Accessed: 1 February 2025).
- **Google Cloud (n.d.) From turnkey to custom**: Tailor your AI risk governance to help build confidence. Available at: Google Cloud (Accessed: 1 February 2025).

- **Google Cloud (n.d.) Getting started with generative AI? Here's how in 10 simple steps. Available at**: Google Cloud (Accessed: 1 February 2025).
- **Google Cloud (n.d.) Optimize prompts | Generative AI on Vertex AI. Available at**: Google Cloud (Accessed: 1 February 2025).
- **IBM (n.d.) Escalations overview - IBM Documentation. Available at**: IBM (Accessed: 1 February 2025).
- **IBM (n.d.) General Data Protection Regulation (GDPR) - Legal Text. Available at**: IBM (Accessed: 1 February 2025).
- **IBM (n.d.) What is compliance monitoring? Available at**: IBM (Accessed: 1 February 2025).
- **IBM (n.d.) What is multimodal AI? Available at**: IBM (Accessed: 1 February 2025).
- **IBM (n.d.) What is NLP (Natural Language Processing)? Available at**: https://www.ibm.com/think/topics/natural-language-processing (Accessed 7 April 2025).
- **IBM (n.d.) What is sentiment analysis? Available at**: IBM (Accessed: 1 February 2025).
- **Intercom (n.d.) The best AI agent built on the best customer service platform. Available at**: Intercom (Accessed: 1 February 2025).
- **LangChain (n.d.) LangChain documentation. Available at**: LangChain (Accessed: 1 February 2025).
- **Microsoft. (n.d.) AI powering customer experience. Available at**: https://news.microsoft.com/europe/features/ai-powering-customer-experience/ (Accessed: 3 February 2025).

- **Microsoft (n.d.) Intent recognition overview - Speech service - Azure AI services. Available at**: Microsoft (Accessed: 1 February 2025).
- **Microsoft (n.d.) What is PII? | Microsoft 365. Available at**: Microsoft (Accessed: 1 February 2025).
- **Microsoft Cloud Blog (n.d.) 5 key features and benefits of large language models. Available at**: Microsoft Cloud Blog (Accessed: 1 February 2025).
- **Microsoft Learn (n.d.) Implement sequential conversation flow - Bot Service. Available at**: Microsoft Learn (Accessed: 1 February 2025).
- **Microsoft Learn (n.d.) Choose effective trigger phrases - Microsoft Copilot Studio. Available at**: Microsoft Learn (Accessed: 1 February 2025).
- **Mordor Intelligence (n.d.) Global Chatbot Market - Growth, Trends, and Forecasts (2023-2028). Available at**: Mordor Intelligence (Accessed: 25 November 2024).
- **NVIDIA (n.d.) Six steps toward AI security. Available at**: NVIDIA (Accessed: 1 February 2025).
- **NVIDIA (n.d.) What is retrieval-augmented generation aka RAG? Available at**: NVIDIA (Accessed: 1 February 2025).
- **OpenAI (n.d.) Fine-tuning - OpenAI API. Available at**: OpenAI (Accessed: 1 February 2025).
- **Pinecone (n.d.) The vector database to build knowledgeable AI. Available at**: Pinecone (Accessed: 1 February 2025).

- **Prompting Guide (n.d.) Chain of Thought (CoT) prompting. Available at**: https://www.promptingguide.ai/techniques/cot (*Accessed 23 April 2025*)
- **Slack (n.d.) Windows | Downloads. Available at**: Slack (Accessed: 1 February 2025).
- **Sviokla, J. (2024) AI Is the New UI. 3 Steps Business Leaders Must Take Now. Forbes. Available at**: Forbes (Accessed: 22 November 2024).
- **TechTarget (n.d.) What is compliance risk? Available at**: TechTarget (Accessed: 1 February 2025).
- **The Go Programming Language (n.d.) Go programming language. Available at**: Go.dev (Accessed: 1 February 2025).
- **Uzoka, A., Cadet, E., and Ojukwu, P.U. (2024) Leveraging AI-powered chatbots to enhance customer service efficiency and future opportunities in automated support. ResearchGate. Available at**: https://www.researchgate.net/publication/385230161 (Accessed: 1 February 2025).
- **Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., and Polosukhin, I. (2017) Attention is all you need. arXiv preprint arXiv**:1706.03762. Available at: https://arxiv.org/pdf/1706.03762 (Accessed: 1 February 2025).