# Story so Far …

CXI evaluates conversations retrospectively, i.e. completed conversations. Every conversation is evaluated using LLMs on 5 independent parameters, and then the score is normalised and combined to calculate the CXI Score.

We generate **scores + justifications** for 5 metrics:

| Metrics | What It Measures | Why It’s Important |
| --- | --- | --- |
| Turn Efficiency Rate | Measures wasted or inefficient chatbot turns (e.g., fallbacks, misclassified intent, unhelpful/vague replies, repetition, apologies). | Ensures conversations are smooth and efficient, reducing customer frustration and effort. Highlights where the bot is wasting turns instead of helping. |
| Topic Shift Rate | Evaluates how well the chatbot detects, adapts to, and manages topic changes. Looks at adaptation quality, relevance of responses after shift, and user engagement. | Conversations often drift or evolve; the bot must adapt seamlessly. Prevents confusion, keeps the user engaged, and shows resilience to real-world usage. |
| Task Completion Rate | Assesses whether the user’s primary task was completed, categorising into fully resolved, partially resolved, or unresolved. | Directly linked to business outcomes and customer satisfaction. If the bot doesn’t help users achieve their goals, the experience fails. |
| Chatbot Intelligence Score | Evaluates overall conversational quality across clarity, naturalness, originality, adaptability, and personalisation. | Captures the “human-like” quality of the interaction. A bot that is clear, natural, and personalised feels more intelligent and builds trust. |
| Sentiment Trajectory Score | Tracks the user’s emotional journey throughout the conversation, categorising the impact of the chatbot on sentiment. | Helps understand how interactions affect customer emotions. Even if a task is completed, a negative emotional trajectory could harm satisfaction and loyalty. |

# Case Studies

## [Confident AI — DeepEval Framework](https://www.confident-ai.com/blog/llm-chatbot-evaluation-explained-top-chatbot-evaluation-metrics-and-testing-techniques)

### **Problem they address**

* Chatbots often work in **multi-turn conversations**, and common models/tests that only evaluate one question → one answer (single-turn) miss lots of important issues (e.g. forgetting context, repeating questions, being inconsistent).
* As conversations get longer, chatbots may produce responses that *feel* polite but are irrelevant, or they may lose or ignore information provided earlier.

### **Solution / Metrics they choose & why**

Confident AI’s DeepEval defines multiple **conversation-level** metrics to evaluate full dialogues (multi-turn), not just last replies. Key metrics they emphasise:

| **Metric** | **What it means** | **Why it matters** |
| --- | --- | --- |
| **Role Adherence** | Does the bot stick to its defined “role” (persona or behaviour instructions) throughout the conversation, across all turns? | Helps ensure consistency, user expectations, trust. If the bot is supposed to be a “customer service agent,” it shouldn't start acting like a casual friend or giving unrelated suggestions. |
| **Conversation Relevancy** | How many of the responses are relevant to the user’s query and the recent context, using a “sliding window” of prior turns to decide context. | Keeps user satisfied; irrelevant or off-topic responses frustrate. Also helps detect when context is lost. |
| **Knowledge Retention** | Can the chatbot remember and use earlier info provided by the user, rather than repeatedly asking for something the user already gave? | Prevents redundancy, improves smoothness of conversation. If a bot asks again for your name or address after you've given it, that’s bad. |
| **Conversation Completeness** | Were the user’s intents or goals fully addressed over the conversation? i.e. after the full conversation, did we cover all the user’s request(s)? | Measures overall effectiveness: it’s possible to have polite, relevant responses but still fail to meet what the user needed. |

*Other metrics they mention: custom metrics you define, and “last-best-response” evaluation (just the final bot response) as a simpler mode, though it trades off detail.*

### Their Approach / How they do it

* Use **“conversational test cases”**: basically, sets of turns (user + bot) collected together to represent a real conversation.
* They offer two evaluation modes:
  1. **Entire conversation evaluation** — look at all turns in the transcript, use metrics that depend on remembering earlier parts.
  2. **Last best response evaluation** — focus only on the last response, but still consider previous context as needed. Simpler and cheaper but less comprehensive.
* Use sliding window technique for context: when evaluating a given turn’s relevance, they look back at a limited number of recent turns to see whether the response makes sense in that context. Too much history is expensive or noisy; too little may miss context.
* Tools: DeepEval (open source) provides metric implementations, test case definitions, ability to benchmark changes, do regression tests, monitor over time.

### Key Take-aways (for us)

* Evaluating full conversation (all turns), not just last response, helps catch issues like **forgetting**, **repetition**, **context loss**.
* Metrics like *Knowledge Retention* and *Conversation Completeness* map closely to good user experience and resolution.
* Role Adherence is important: how well the bot adheres to its intended purpose.

# 2. [Microsoft Article](https://medium.com/data-science-at-microsoft/evaluating-llm-based-chatbots-a-comprehensive-guide-to-performance-metrics-9c2388556d3e)

### **Problem They’re Solving**

LLM-based chatbots are powerful but *hard to evaluate*, because:

* Conversations are free-form, multi-turn, and varied in style. It’s not enough to check just the final reply.
* Explicit user feedback (e.g. surveys or thumbs up/down) is often rare.
* Need to ensure responses are accurate, relevant, not misleading or hallucinating.
* Also need evaluation methods that scale, that can be automated, and that respect privacy.

### **Metrics They Choose & Why**

Microsoft proposes a set of metrics, grouped by theme. Here are some of their main ones:

| **Metric** | **What It Measures** | **Why It’s Important** |
| --- | --- | --- |
| **Search Performance (for RAG systems)** | How well the retrieval component works: relevance of fetched documents; is the same result retrieved when queries are paraphrased. | Because if retrieval fails, the LLM will have no good source to base its answer. It ensures domain-knowledge / updated info is included. |
| **Task Completion** | Whether the chatbot fulfils the user’s request or goal. | Core measure of success: the user wants something done; if not done, the conversation has failed in its purpose. |
| **Intelligence** | How “smart” or impressive the responses are: creativity, insight, adaptability, originality. | Helps distinguish between a bland, correct answer and one that feels like it adds value/user delight. |
| **Relevance** | Are responses appropriate to context, tone, user’s query, clarity, courteous? | Even if a response is correct, if it's confusing, rude, or off-tone, it hurts user experience. |
| **Hallucination** | Whether the response invents facts, or isn't grounded in truth or retrieved data (for RAG). | In finance/regulated settings, hallucinations can be dangerous (misinformation, risk). |
| **No Response / Fallbacks** | Measures how often the chatbot fails to answer meaningfully: either says “I don’t know,” or gives generic fallback apologies. | Too many fallbacks = poor coverage, user frustration. |
| **Responsible AI / Ethics** | Monitoring when responses or user prompts are blocked, or when the model risks violating policy. | Ensures chatbot behaves safely, ethically, and compliant with legal / company rules. |
| **Resource Utilization** | How efficient is the system in terms of token usage, unnecessary retrieval, etc? | Helps with cost, latency, or computational waste. Important in large-scale deployment. |
| **User Feedback & Sentiment** | Even if explicit feedback is missing, infer satisfaction via sentiment analysis, downstream behaviour (follow-ups, repeat queries, etc.). | Gives signals about real user happiness (or frustration), helps detect regressions unseen by technical metrics. |
| **User Intent / Domain Categorisation** | Recognising which domain or use case the user is in (e.g. banking, support, or troubleshooting) to understand what metrics or expectations apply. | Because different intents may have different success criteria, it helps break down performance by domain. |

### **Their Approach / How They Do It**

* Use both **retrieval-augmented generation (RAG)** and pure generation; when RAG is used, they evaluate both the retrieval component and the generation.
* For relevance/robustness: check **search stability** (do similar queries get similar results), **search relevance** (how good are retrieved documents' rank/quality) using metrics like NDCG, etc.
* Monitor **fallback/apology/no response** rates via log telemetry.
* Use implicit signals: e.g. follow-up questions, user engagement, sentiment, copying responses, link clickthroughs when explicit feedback is missing.
* Always keep **a human in the loop** for auditing: sample conversations for manual check.

### **What’s Good / Risks They Highlight**

**Pros:**

* Broad set of metrics → covers many failure modes (accuracy, relevancy, ethics, realism).
* Scalable: many metrics are automated.
* Makes up for lack of explicit feedback using implicit signals.

**Risks / Challenges:**

* Quality of the retrieval base matters; if the knowledge base is poor, “hallucination” metrics suffer.
* Some metrics are subjective (e.g. “intelligence”) — need well-designed prompts and human benchmarks.
* Hidden costs: monitoring, log storage, audit, and computing resources.
* Potential bias: tone, domain might favour certain user types; must monitor fairness.

### Takeaways / What We Could Apply

* Strong case for adding **Hallucination** as a metric, especially since banking demands factual correctness.
* Evaluate *fallback / no response* rates to catch where the bot fails outright.
* Use **implicit signals** (follow-ups, sentiment, link clicks, etc.), especially when we don’t have explicit feedback.
* Perhaps split “Chatbot Intelligence” into sub-parts: intelligence, relevance, clarity, etc., like Microsoft’s “intelligence + relevance + coherence + tone” mix.
* If we ever use retrieval or a knowledge base, evaluate the retrieval part (search relevance, stability) separately.
* Use log data/telemetry to capture “no response” / fallback / apology rates.

## [**Instacart LACE — Summary**](https://tech.instacart.com/turbocharging-customer-support-chatbot-development-with-llm-based-automated-evaluation-6a269aae56b2)

### **What Problem They Are Solving**

* The need to **measure chatbot quality in real conversations** (multi-turn, messy, real user issues) rather than only testing in lab settings.
* Hard to know if changes to the bot are improving the customer experience, or which failures are most important to fix.
* Want more reliable, consistent evaluation that mirrors human judgment but that can scale.

### **The Solution: LACE (LLM-Assisted Chatbot Evaluation)**

* **Five Key Dimensions** that each conversation is judged on:  
  1. **Query Understanding** — did the bot properly understand what the user was asking?
  2. **Answer Correctness** — are answers factually correct, relevant, consistent, and useful?
  3. **Chat Efficiency** — are there wasted or repetitive turns; is the bot being concise and efficient?
  4. **Client Satisfaction** — does the conversation feel satisfying from the user’s perspective?
  5. **Compliance** — is the response aligned with policies, terms, and risk/regulatory-conscious behaviors?
* Each criterion inside these dimensions is binary (True / False). E.g. for Answer Correctness, there are criteria like “contextual relevancy,” “factual correctness,” etc.
* For each conversation session (the full multi-turn interaction), they compute whether each criterion passed or failed; then they aggregate to generate an overall quality score for that session.

### **Their Approach: How LACE Works**

* They compare and test **three evaluation methods**:  
  1. **Direct Prompting** — single-pass prompt: the LLM evaluates based on the criteria in one go.
  2. **Reflection** — LLM evaluates, then reflects on its own evaluation and possibly revises it.
  3. **Debate-style Agents** — multiple LLM agents take different roles (customer critic, support defender, judge) and debate or review the conversation; the judge aggregates.

* They decided on binary scoring rather than fine-grained scales, because binary tends to be more consistent and aligns better with human judgment, and is simpler to maintain.
* They separate **reasoning / explanation (free-text)** from the structured output (e.g. JSON). First the LLM gives its reasoning, then that is parsed or converted into structured fields like “True/False for each criterion + explanation.”
* They also iteratively validate: they sample human evaluators on transcripts, compare human vs LACE judgments, refine criteria, refine prompts, then repeat until alignment is good.

### **Why Those Metrics / Design Choices**

* The five dimensions cover different facets of user experience: understanding, correctness, efficiency, satisfaction, and compliance. Together they give a holistic picture.
* Binary criteria reduce ambiguity and improve consistency across evaluators / sessions. Easier for LLM prompts.
* Reflection / debate approaches help capture nuance and reduce mistakes (LLM missing a detail, getting misled by prompt, etc.)
* Separating reasoning from structured output helps maintain clarity and flexibility: reasoning lets you see “why” a score was given, structured output makes machine-processing / dashboards easier.
* Pilot studies + human alignment ensure that automated scoring does not drift away from what humans think “good” looks like.

### What We Can Learn / Apply

Here are direct lessons we might apply to our evaluation framework:

| **Feature from Instacart LACE** | **Why It Could Help Us** | **Potential Implementation** |
| --- | --- | --- |
| Use of **binary criteria** per dimension (True/False) | Simplifies scoring; improves consistency; easier to interpret failures. | Convert some of our metrics into binary checks for key sub-criteria, e.g. “Did the bot understand user intent?” Yes/No. |
| Multi-step evaluation (Reflection, Debate) | More nuanced assessments; reduces errors; catches subtle failures. | Prototype a reflection pass or debate agents for difficult chats or edge cases. |
| Free-form reasoning + structured outputs | Makes the reasoning transparent; supports auditing; helps with traceability. | Ensure our system stores the LLM rationale and also structured JSON (score + explanation) per metric. |
| Human alignment & iteration | Ensures that our automated scores reflect actual customer / human judgments; prevents drift. | Set up regular calibration sessions: humans rate a sample, compare with LLM’s results, refine prompts / criteria. |
| Focus on efficiency & client satisfaction, not just correctness | Having correct answers is necessary but not sufficient: if the bot is sluggish or repetitive, customers still suffer. | Keep “efficiency” and “client satisfaction” (or sentiment) as first-class metrics; monitor things like repeated or wasted turns. |

# **CXI Evaluation Framework — Benchmarking & Best Practices**

## **Our Approach (CXI)**

For CXI, we use **LLM-based retrospective evaluation** to score chatbot conversations once they are complete.

* Each conversation is scored using **five independent metrics**.
* For each metric, the LLM produces both a **numeric score** and a **justification**.
* The scores are normalised and combined into an overall **CXI Score**, which represents the customer’s experience of the chatbot.

### **Our Metrics**

| **Metric** | **What It Measures** | **Why It’s Important** |
| --- | --- | --- |
| **Turn Efficiency Rate** | Identifies wasted or inefficient chatbot turns (e.g., fallbacks, vague replies, repetition, unnecessary apologies). | Smooth and efficient conversations reduce frustration. Inefficiency highlights where the bot fails to move the conversation forward. |
| **Topic Shift Rate** | Assesses how well the chatbot detects, adapts to, and manages topic changes, including the relevance of replies after the shift. | Real customer interactions often drift between topics; successful bots adapt seamlessly without confusing the user. |
| **Task Completion Rate** | Checks whether the user’s task was fully resolved, partially resolved, or not resolved. | Directly tied to business outcomes: if a customer can’t complete their task, the experience fails. |
| **Chatbot Intelligence Score** | Evaluates clarity, naturalness, adaptability, originality, and personalisation of the bot’s replies. | Builds trust and engagement by making the bot feel more “human-like” and reliable. |
| **Sentiment Trajectory Score** | Tracks the user’s emotional journey, showing whether the bot improved, worsened, or maintained customer sentiment. | Ensures interactions don’t harm satisfaction or loyalty, even if tasks are completed. |

## **Case Studies — Industry Practices**

### **1. Confident AI — DeepEval Framework**

* **Problem**:  
   Traditional evaluation focused only on single-turn QA pairs. This misses problems like:  
  + Bots forgetting earlier context,
  + Repeating questions unnecessarily,
  + Going off-topic while still sounding polite.
* **Solution**: Multi-turn evaluation through **DeepEval**.  
  + **Role Adherence**: Ensures the bot consistently acts like the defined persona (e.g., banking assistant vs. casual friend).
  + **Conversation Relevancy**: Checks if responses remain relevant using a sliding window of recent context.
  + **Knowledge Retention**: Measures if the bot remembers earlier information (avoiding asking again for details like name or account).
  + **Conversation Completeness**: Ensures all user intents and goals were addressed by the end.
* **Approach**:  
  + Evaluate entire conversations, not just the last response.
  + Use sliding windows for context awareness.
  + Provide two evaluation modes: **entire transcript** and **last-best-response** for cost efficiency.
* **Why It Matters**: Real conversations are messy; success requires retention, relevance, and completeness across all turns.

### **2. Microsoft — Comprehensive Evaluation Framework**

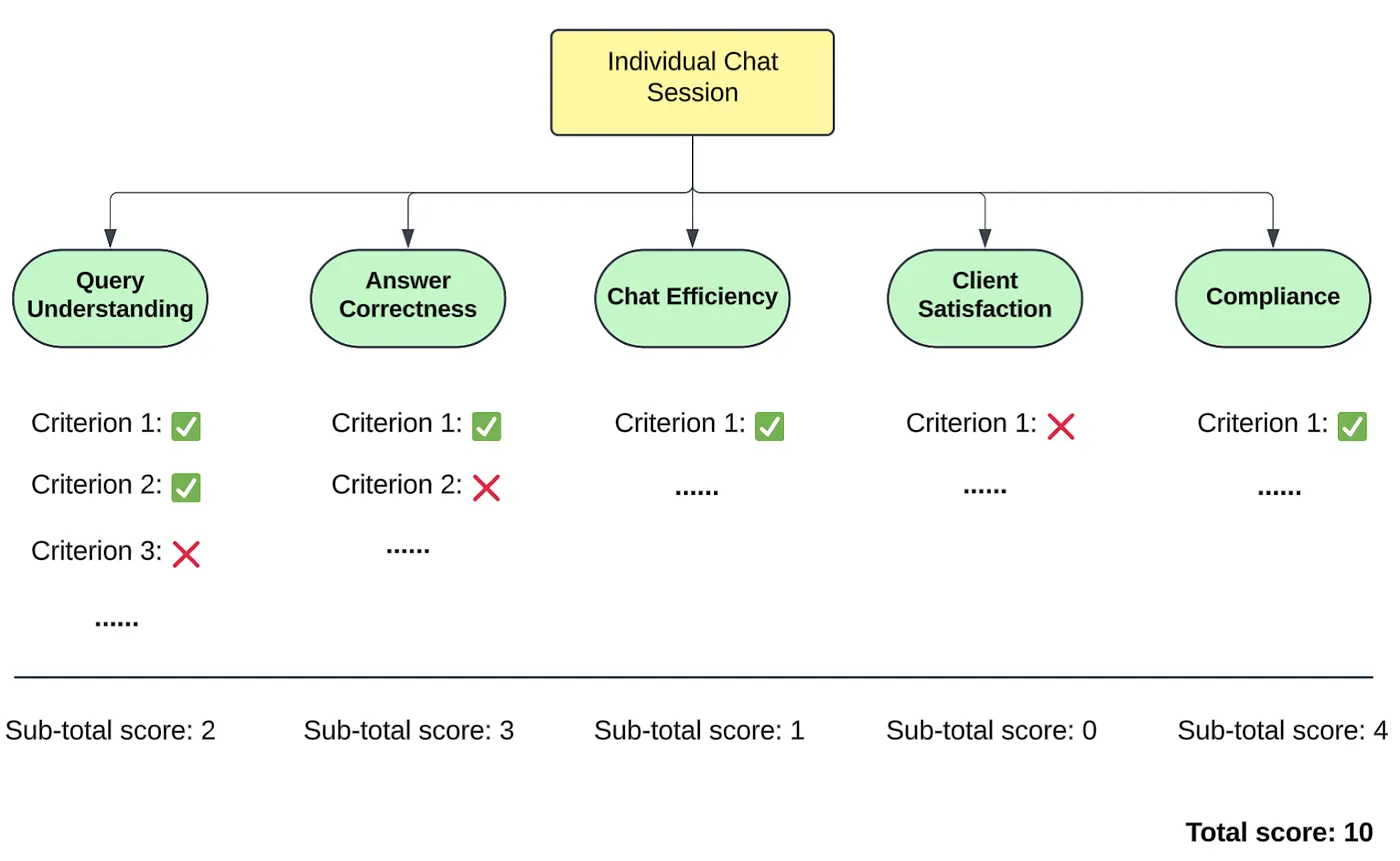
* **Problem**:  
  + Chatbot conversations are free-flowing, unpredictable, and multi-turn.
  + Explicit feedback (surveys, thumbs-up/down) is rare.
  + Risk of **hallucination** and **ethical issues** in high-stakes industries.
* **Solution / Metrics**:  
   Microsoft groups metrics into categories:  
  + **Core Performance**: Task completion, relevance, intelligence.
  + **Truthfulness**: Detect hallucinations or fabricated answers.
  + **Reliability**: Track fallback/no-response rates.
  + **Ethics & Safety**: Ensure compliance with policies, fairness, and Responsible AI.
  + **System Health**: Monitor retrieval quality (for RAG systems) and efficiency.
  + **User Signals**: Sentiment shifts, repeat queries, follow-ups, engagement metrics.
* **Approach**:  
  + Evaluate **retrieval** and **generation** separately.
  + Use **implicit signals** (clicks, follow-ups) where explicit feedback is missing.
  + Human auditors sample transcripts for quality checks.
* **Why It Matters**: Covers both customer experience and business risk, which is crucial for regulated industries like banking.

### **3. Google — Conversational Insights & Quality AI**

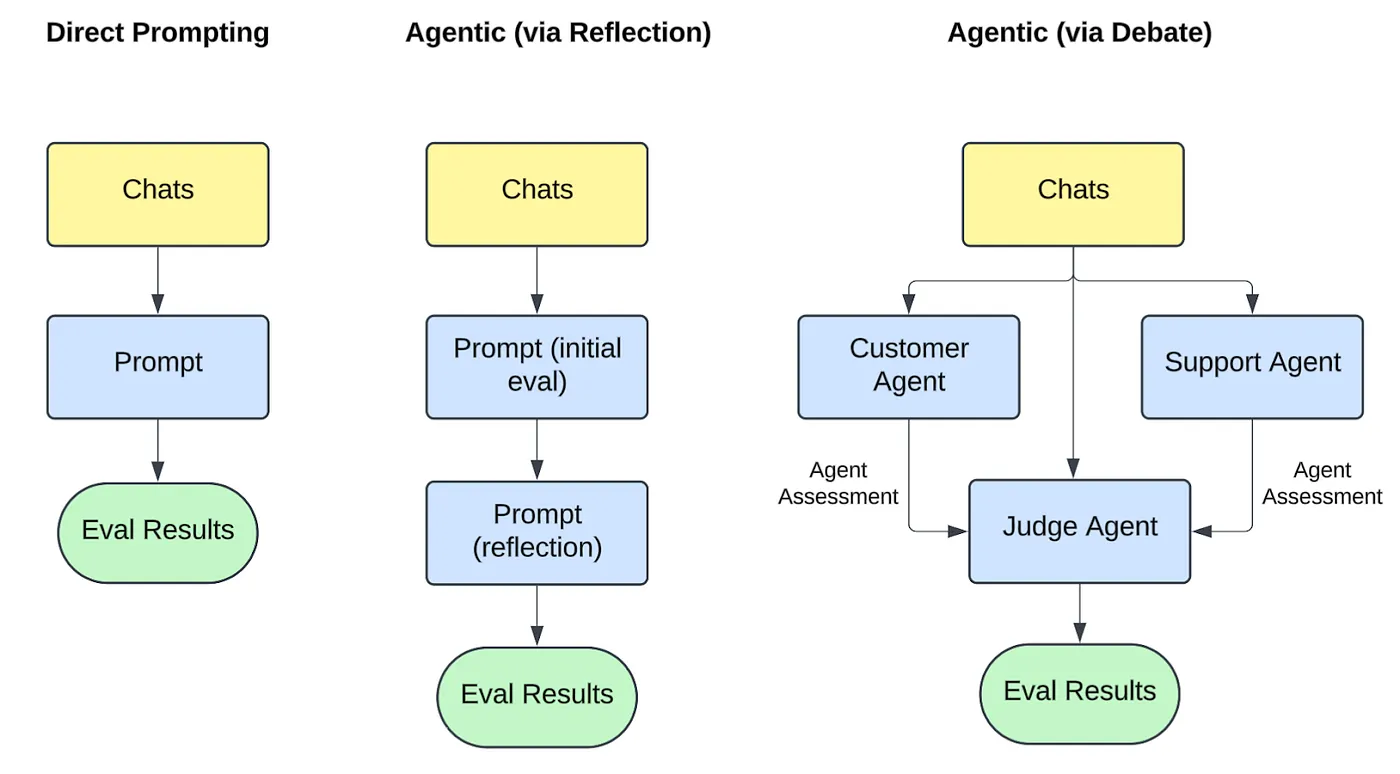
* **Problem**:
  + Human QA in contact centres is inconsistent and doesn’t scale.
  + Businesses need consistent scoring to improve service quality.
* **Solution / Metrics**:
  + **Quality AI** uses LLMs to automatically score entire conversations.
  + Measures: **intent recognition, empathy, compliance, helpfulness, and resolution**.
* **Approach**:
  + Generates both **quality scores** and **justifications** (text explanations).
  + Dashboards track trends across agents and conversations.
  + Designed for **scalable enterprise QA** across thousands of interactions.
* **Why It Matters**: Goes beyond task success to include **soft skills** like empathy and compliance — critical in customer-facing contexts.

### **4. Instacart — LACE (LLM-Assisted Chatbot Evaluation)**

* **Problem**:  
   Needed a scalable, reliable chatbot evaluation for **messy, real-world multi-turn conversations**.
* **Solution / Metrics**:  
  + **Query Understanding** → Did the bot grasp user intent?
  + **Answer Correctness** → Are responses factually correct and relevant?
  + **Chat Efficiency** → Are there wasted or repetitive turns?
  + **Client Satisfaction** → Did the user feel supported?
  + **Compliance** → Are responses aligned with policies and regulations?
* **Approach**:  
  + **Binary Scoring (True/False)** per dimension for consistency.



* + Tried three methods: **single-pass prompts**, **reflection**, and **debate-style evaluation**.
  + Separate **reasoning** (LLM explanation) from **structured JSON output**.
  + Validated against human evaluators for alignment.



* **Why It Matters**: Binary metrics increase reliability, reflection/debate reduces errors, and human alignment ensures trustworthiness.

## **Parameters Suggested by Studies (From Conversations Only)**

Since explicit survey data is often missing, studies recommend extracting **user experience signals directly from conversation logs**. These can be computed with LLMs or log-based heuristics:

* **Task Success** → Resolution status, clarity of outcome (Walker et al., 1997).
* **User Effort** → Turns to resolution, fallback rate, clarification requests (Li et al., 2016).
* **Frustration & Emotion** → Sentiment trajectory, escalation requests (Sharma et al., 2020).
* **Answer Quality** → Helpfulness, hallucination/factuality checks, conciseness (Yeh et al., 2021; Huang et al., 2023).
* **Coherence** → Context retention, logical flow, consistency across turns (DeepEval, Microsoft).
* **Politeness & Empathy** → Detection of acknowledgement, reassurance, politeness markers (Sharma et al., 2020).
* **Compliance & Safety** → Adherence to regulatory and ethical standards (Microsoft; Google).
* **Knowledge Retention** → Bot remembers user-provided details, avoids re-asking (Confident AI).
* **Predicted CSAT (pCSAT)** → Predict user satisfaction scores directly from conversation features and sentiment trajectory (Zhang et al., 2024).

## **Combined Learnings & Takeaways**

| **Source** | **Key Strengths** | **Lessons for CXI** |
| --- | --- | --- |
| **Confident AI** | Multi-turn focus: role adherence, relevancy, retention, completeness. | Add Knowledge Retention and Role Adherence. |
| **Microsoft** | Broad coverage: hallucination, fallbacks, implicit signals, and ethics. | Add Hallucination Check, Fallback Rate, and implicit signals (follow-ups, repeats). |
| **Google** | Empathy, compliance, and resolution at scale. | Strengthen Empathy/Politeness and Compliance monitoring. |
| **Instacart** | Simple binary checks + reflection/debate + human alignment. | Introduce binary sub-metrics, test reflection/debate methods, and calibrate with human reviews. |
| **Academic Research** | PARADISE links success to efficiency; learned metrics (e.g., BERTScore, BLEURT, USR) better capture relevance; hallucination detection & empathy modelling are critical; CSAT can be predicted from transcripts. | Incorporate Factuality/Hallucination checks, User Effort indices, Predicted CSAT, and Empathy/Helpfulness proxies. |

## **CXI vs. Industry & Academia**

✅ **Already Strong In:**

* Retrospective evaluation of entire conversations.
* Multi-dimensional scoring (efficiency, task success, sentiment, intelligence).
* Scores + justifications (like Google & Instacart).

⚡ **Areas to Improve (from research + industry):**

* **Hallucination & Factual Accuracy** → Microsoft, Instacart, Huang et al. (2023).
* **Knowledge Retention & Role Adherence** → Confident AI, Walker et al. (1997).
* **Compliance & Empathy** → Google, Sharma et al. (2020).
* **Binary Sub-Metrics for reliability** → Instacart.
* **Implicit Feedback Integration** → Microsoft; CSAT prediction studies.
* **Reflection/Debate Methods** → Instacart.
* **Predicted CSAT (pCSAT)** → Academic work on conversation-based CSAT prediction.

## **References**

* Walker, M. et al. (1997). **PARADISE: A Framework for Evaluating Spoken Dialogue Agents**. ACL.
* Li, J. et al. (2016). **Deep Reinforcement Learning for Dialogue Generation**. ACL.
* Sharma, A. et al. (2020). **Computational Approaches to Empathy in Dialogue Systems**.
* Yeh, Y.-T. et al. (2021). **Assessment of Automatic Evaluation Metrics for Dialogue**.
* Huang, L. et al. (2023). **A Survey on Hallucination in Large Language Models**.
* Recent studies (2024–2025). **Predicting Customer Satisfaction by Replicating the Survey (pCSAT methods)**.
* Confident AI. (2023). **DeepEval Framework**.
* Microsoft. (2024). **Evaluating LLM-based Chatbots: Comprehensive Guide to Performance Metrics**.
* Google Cloud. (2023). **Conversational Insights and Quality AI**.
* Instacart Tech Blog. (2024). **Turbocharging Customer Support Chatbot Development with LLM-based Automated Evaluation**.
* Zhang et al., 2024. Predicting Customer Satisfaction from Dialogue Logs