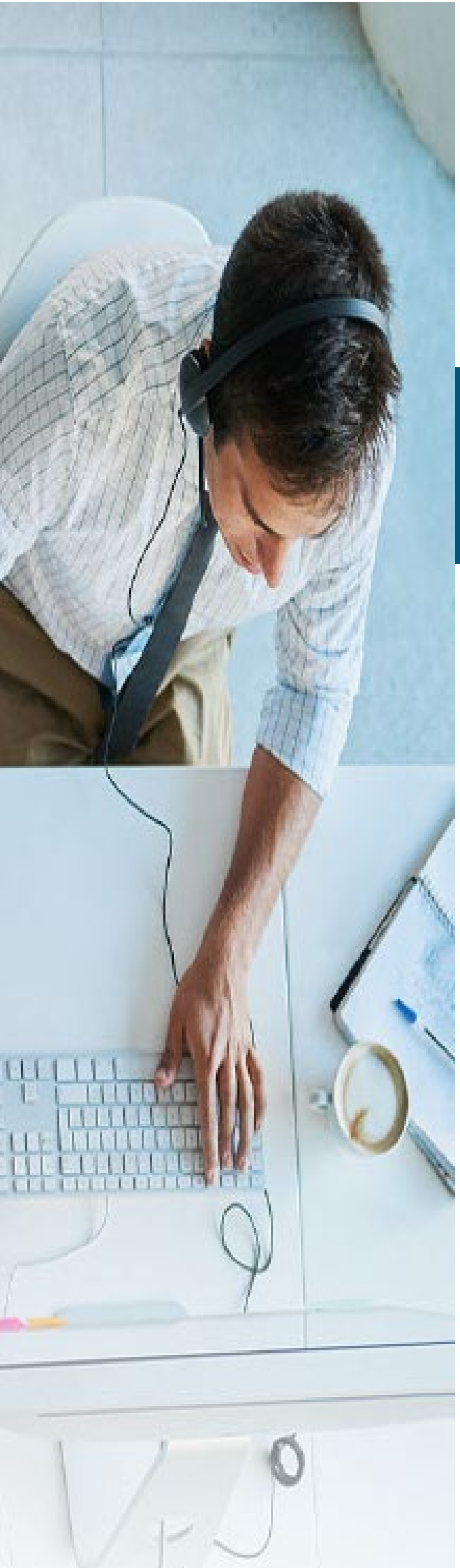

CALL RECORDING

Conversation Analytics 101

AI Tools

Admin Guide

MOMENTUM

Powered by: miarec

1. Guide Overview

This guide explains ways authorized Call Recording Administrators can use Call Recording Conversation Analytics (AI) tools for tasks like: enabling call transcription, enabling AI insights, validating results, and rolling out dashboards and workflows.

> **New to Conversation Analytics?**
> Start your learning by reading the *CR: Conversation Analytics (AI) 101 QRG*.

1.1 Who this guide is for

- **Tenant administrators** of Service Provider-hosted deployments

1.2 What you can configure (tenant view)

As a tenant admin, you typically control:

- **Transcription enablement and validation** (for voice calls)
- **Custom Fields** used to store AI insights (scores, categories, extracted values)

AI Tasks:

- enable/disable prebuilt tasks
- configure or create tenant-specific tasks
- override prompts and filters (where permitted)
- **Testing and validation** using Playground / “Save and Test”
- **Dashboards, search filters, and reporting** to make insights actionable
- **Permissions/visibility** (role access and field visibility where supported)

1.3 What this guide does not cover

- System/global configuration of engines, global tasks, user management tasks, background jobs, non-AI related administration tasks, etc.

1.4 How this guide is organized

1. **Core Configuration** – transcription, fields, tasks, overrides, Auto QA
2. **Testing and Tuning** – Playground + prompt best practices
3. **Dashboards and Search** – how metrics become dashboards and filters
4. **Monitoring and Troubleshooting** – common issues and fixes
5. **Insight Types** – helpful information about commonly used insight types

1.5 Recommended initial rollout path (practical)

If you're implementing Conversation Analytics for the first time, follow this sequence:

1. **Enable and validate transcription** (voice calls)
2. **Enable 2–3 prebuilt AI Tasks** (e.g., Summarization + Sentiment + CSAT)
3. **Test results on real conversations** using Playground
4. **Confirm results appear** in Conversation Details, Dashboards, and Search
5. **Roll out dashboards and saved searches** to supervisors and QA teams
6. Expand with **custom insights** and **Auto QA** as needed or desired ($$)

2. Requirements

This chapter lists the requirements for enabling and using Conversation Analytics (AI) for your tenant.

2.1 Access and permissions

You need permissions to:

- Manage **Custom Fields**
  `Administration > Customization > Custom Fields`
- View and manage **AI Tasks**
  `Administration > Speech Analytics > AI Assistant > AI Tasks`
- Access **Playground / testing** tools (if your deployment exposes them)
- Manage **dashboards** and **reports** (if dashboards are permissioned)

Typically, a **Tenant-level Admin** role is required for managing Custom Fields and AI Tasks.
Role names and permissions may vary by deployment. Pleasecontact your service provider with questions about specific role requirements.

2.2 Data prerequisites

Conversation Analytics requires:

Conversation content

- Calls: audio + metadata
- OPTIONAL: Chats/emails/tickets: text + thread metadata (as supported in your deployment)

Transcripts/threads

- Voice calls must be transcribed before AI insights can run
- OPTIONAL: Text channels must have normalized message threads available

2.3 Operational prerequisites (tenant view)

Depending on deployment model:

- In **partner-hosted** deployments, your service provider may expose specific settings, but tenant-level Admins will focus on management of those exposed/displayed tenant-level configuration and settings.

Tenant-level admins typically have access to AI tools for:

- **AI Tasks** – enable, disable, and override tasks
- **Playground / testing** – test prompts on sample conversations (via "Test" button on task view)

2.4 Supported configuration types (AI insights)

AI Tasks can populate Custom Fields that are commonly:

- Numeric (e.g., CSAT 1–5, lead score)
- Dropdown classification (e.g., churn risk: low/medium/high)
- Date/time extraction (e.g., reservation start date)
- Free text (e.g., summary, next actions)
- Multi-field outputs (one task writes several fields)

3. Getting Started Checklist

This checklist can be used to plan and launch Conversation Analytics (AI) quickly and safely. It assumes you are a **tenant-level administrator**.

3.1 Phase 1 — Verify requirements are met (30–60 mins)

- Confirm you can access **Custom Fields**
  `Administration > Customization > Custom Fields`
- Confirm you can access **AI Tasks**
  `Administration > Speech Analytics > AI Assistant > AI Tasks`
- Confirm you have sample conversations with **transcripts** (for calls) or **message threads** (for text channels)
- Confirm you have at least one internal stakeholder for validation (QA lead, CX leader, Sales manager)

3.2 Phase 2 — Enable transcription (voice calls)

- Enable and validate transcription (see: **Enable Transcription (Prerequisite)**)
- Spot-check 10–20 recent calls for transcript quality and language correctness

> If you do not see transcription settings in your tenant: Your organization did not order Transcription services or Transcription may be managed by your service provider. Use the validation checklist in the transcription chapter and contact your provider if changes are required.

3.3 Phase 3 — Enable “starter” AI Tasks

In `Administration > Speech Analytics > AI Assistant > AI Tasks`:

- Open the **Disabled** tab
- Enable these starter tasks (recommended):
- **Conversation Summarization**
- **Sentiment Analysis**
- **CSAT** (if you want CX scoring right away)

> You can start with only one task. Fewer tasks = easier validation and lower cost.

3.4 Phase 4 — Test and calibrate before broad rollout

- Use **Playground** or **Save and Test** to validate each enabled task on 5–10 representative conversations
- Confirm outputs are:
- present (not blank)
- sensible (value matches transcript)
- consistent with your definitions
- accompanied by a short explanation (where configured)

If results don’t match your expectations:

- Override **Prompt**, **Filter**, or both (see: **Override Prompt and Filters**)
- Re-test until you reach acceptable consistency

3.5 Phase 5 — Make insights actionable

- Confirm insights appear in **Conversation Details** (Analytics)
- Configure/verify **dashboards** for key metrics (CSAT, churn risk, etc.)
- Create **saved searches** for operational workflows:
  - Low CSAT calls (e.g., CSAT < 3)
  - High churn risk calls
  - Escalations by reason

- Inform QA Supervisors about your setup and how to interpret explanations, drill down, etc.

3.6 Phase 6 — Expand or add more and test

- Add additional prebuilt tasks (topics, reason/outcome, sales insights)
- Create **tenant-specific custom insights** (your business-specific fields)
- Configure **Auto QA** if you have a QA program

4. Core Configuration

4.1 Enable Transcription (Prerequisite)

Transcription is required for **voice calls**. AI insights (CSAT, sentiment, summaries, topics, etc.) are produced by analyzing the **call transcript**—if a call has no transcript, AI Tasks will not run for that call.

> For chats/emails/tickets (text channels), transcription may not apply. Instead, Call Recording analyzes the message thread directly (as supported in your deployment).

What you should accomplish in this chapter

- Confirm transcription is enabled for your tenant (or understand if it is provider-managed)
- Validate transcript quality on a sample of recent calls
- Understand how transcription settings affect AI insights (language, diarization, etc.)
- Know what to do when transcripts are missing

Where to configure transcription

Transcription is typically managed at the platform level by your provider. Tenant admins can validate that transcription is working correctly by checking recent call recordings.

> If transcription is not enabled for your tenant or you need to change language settings, contact your provider.

Step 1 — Confirm transcription is enabled

1. Open a recent call in the **Calls** list and check if a transcript is present.
2. If transcription settings are available in your tenant (`Administration > Speech Analytics > Transcription`), confirm transcription is **enabled** for call recordings.
3. Verify any language settings (if visible):
4. auto-detect vs fixed language(s)

5. multilingual tenant rules (if supported)

> **Note:** If transcription settings are not visible in your tenant, transcription is managed by your provider. Contact them if you need to enable transcription or change language settings.

Expected result

New calls should show a transcript in the conversation/call details view shortly after the call is ingested.

Step 2 — Validate transcript quality

Transcript quality impacts AI insight quality. Spot-check at least **10–20** calls across different agents/queues.

Check for: - Correct language and encoding - Speaker attribution (agent vs customer), if supported - Missing segments or heavy distortion - Correct punctuation (helps summarization and extraction)

If transcript quality is poor

- If you manage transcription settings: adjust language/engine settings and re-test.
- If transcription is provider-managed: collect examples and contact your provider.

Step 3 — Understand how transcription affects AI Tasks

AI Tasks are typically configured with filters such as: - inbound only - minimum duration (e.g., > 15 seconds)

Even if transcription is enabled, AI Tasks may not run if: - the call is too short (filtered out) - the call direction/type is excluded by filters - the transcript is missing or delayed

Troubleshooting: transcripts missing

If transcripts are not appearing:

1. Confirm calls are present in Call Recording and playable.
2. Check whether transcripts are delayed vs missing (new calls may take time).
3. Validate that the call language is supported and configured properly.
4. If you recently enabled transcription, confirm whether it applies to:
5. new calls only, or
6. historical calls (backfill)

By default, only new calls are transcribed after transcription is enabled. Historical calls are not automatically backfilled.

If you need transcripts for historical calls, contact your provider to request an on-demand backfill job. Backfill jobs may have limits on date range and volume.

4.2 Custom Fields for Insights

Custom Fields are where Call Recording stores AI insight outputs (scores, categories, extracted values). Once a field is populated, it becomes usable in:

- **Conversation details** (Analytics view)
- **Dashboards** (averages, distributions, trends, drilldowns)
- **Search and filters** (e.g., CSAT < 3)
- **Reports/exports** (where enabled)

Menu path

```
Administration > Customization > Custom Fields
```

When to create a new Custom Field

Create a Custom Field when you want to:

- capture a new business-specific metric (e.g., “Reservation Start Date”)
- store a score (e.g., CSAT, lead score)
- store a classification (e.g., churn risk: Low/Medium/High)
- store extracted entities or text (e.g., “Next actions”)

If a prebuilt AI Task already maps to a prebuilt field, you may not need to create anything—just enable the task.

Field type selection (practical guidance)

Choose the field type based on how you plan to use the value:

- **Number (integer/decimal)** — scoring, averages, thresholds, range filters
  Examples: CSAT (1–5), Lead Score (0–100)
- **Dropdown / choice** — consistent categorization and easy dashboards
  Examples: Churn Risk (Low/Medium/High), Call Outcome
- **Date** — extracted dates for timeline analysis
  Examples: Reservation Start Date, Follow-up date
- **Text** — summaries, reasons, notes, explanations
  Examples: Call summary, “CSAT explanation”, “Top objection detail”

> Recommendation: prefer **Dropdown** over free text for categories you want to chart consistently.

Step-by-step: create a field for AI Insights

1. Go to `Administration > Customization > Custom Fields`.
2. Click **Add** (or equivalent).
3. Configure:
4. **Name** (human-readable)
5. **Computer name** (stable identifier; avoid spaces; don’t change after rollout)
6. **Description** (what the field represents)
7. **Type** (Number / Dropdown / Date / Text)
8. Enable AI Insights integration:
9. Check the **AI Insights** checkbox to make the field available to AI Tasks
10. Configure display:
11. Choose a **Display group** (e.g., "CX Metrics", "Sales Metrics") so the value appears in Conversation Details → Analytics.
12. Save the field.

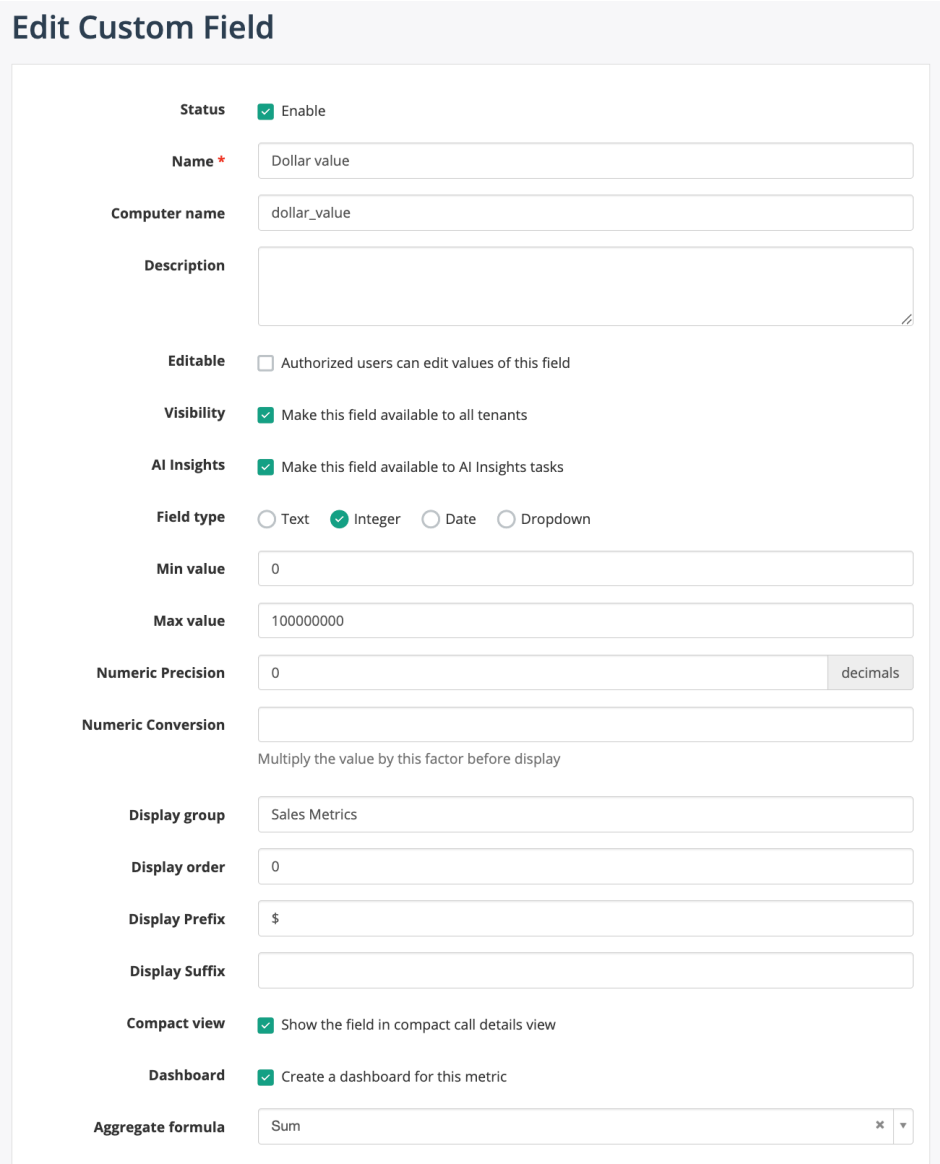

Figure: Custom Field configuration showing AI Insights checkbox and Dashboard option.

Thresholds and labels (scores like CSAT)

For numeric scoring fields, thresholds help you:

- show labels (e.g., “Dissatisfied”)
- apply colors for quick scanning
- create clickable buckets in dashboards

Example CSAT thresholds (1–5): - 1 = Very Dissatisfied - 2 = Dissatisfied - 3 = Neutral - 4 = Satisfied - 5 = Very Satisfied

> Keep thresholds consistent across teams to avoid KPI confusion.

Creating dashboards from fields (recommended for key metrics)

Enable the **Dashboard** checkbox when creating the field to automatically generate a dashboard. This creates:

- average score display
- distribution buckets (based on configured thresholds)
- trend over time
- clickable drilldown to matching conversations

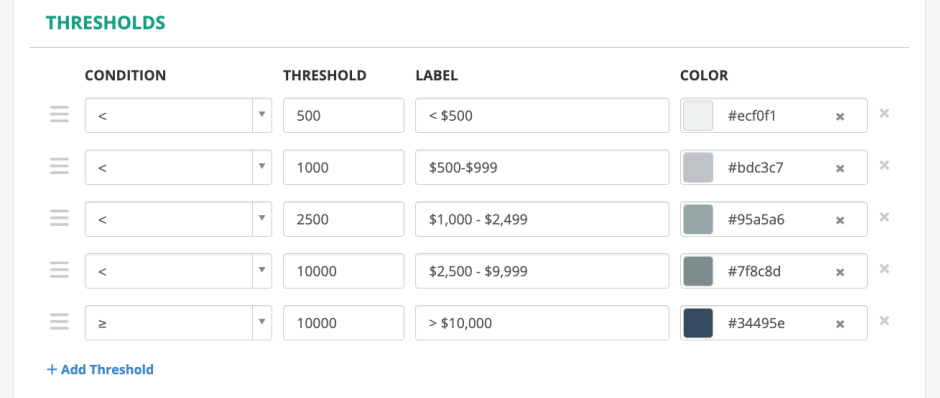

Figure: Threshold configuration for numeric fields. Define buckets with labels and colors for dashboard visualization.

Storing "value + explanation"

Many AI insights are best represented as: - a structured **value** (score/category/date) - plus an **explanation** for a human reviewer

Call Recording stores explanations automatically alongside the insight value. When you configure an AI Task to return both a value and an explanation, the explanation is displayed in the Conversation Details view next to the metric value without requiring a separate custom field.

4.3 AI Tasks: Enable and Manage

An **AI Task** is a purpose-specific analysis definition that reads a conversation transcript/thread and writes one or more outputs into **Custom Fields**.

In Call Recording, an AI Task typically includes:

- **Prompt** (instructions to the AI model, including output format)
- **Attribute mapping** (which output attribute populates which Custom Field)
- Optional **filters** (which conversations the task applies to)
- **AI engine** selection (LLM provider/model), depending on deployment

Menu path

`Administration > Speech Analytics > AI Assistant > AI Tasks`

In this view, tenant admins can usually:

- See tasks that are **Enabled** for the tenant
- Enable tasks from the **Disabled** tab
- Edit tasks to override **Prompt**, **Filter**, or both (where permitted)
- (Optionally) create tenant-specific tasks, if enabled in your deployment

What to check in an AI Task (before enabling)

When you open an AI Task, verify:

1. **Purpose** – the task name/description matches the business outcome you want (e.g., “CSAT Scoring”)
2. **Outputs** – which fields it writes to
3. **Output type** – JSON is recommended for structured metrics
4. **Filters** – whether it applies to the right subset of conversations (e.g., inbound calls only)
5. **Explanation behavior** – whether it includes a reviewer-friendly explanation (recommended)

Mapping: output attributes → Custom Fields

The mapping table connects: - **ATTRIBUTE**: the output key produced by the AI (e.g., `csat`, `churn_risk`) - **CUSTOM FIELD**: where the value is stored for reporting/search/dashboards

One task may populate **multiple fields**, which is useful for: - grouping related outputs (e.g., "Churn Risk" + "Churn Risk Reason") - producing both **value** and **explanation** (if explanations are stored in fields)

Filters: controlling eligibility

Tasks can include filters such as:

- channel (call vs chat vs ticket)
- direction (inbound vs outbound calls)
- duration threshold (e.g., > 15 seconds)
- other metadata-based filters (queues, tags, etc.)

Filters are important for: - relevance (avoid scoring irrelevant calls) - accuracy (avoid scoring too-short conversations) - cost control (reduce unnecessary processing)

Prompt: what you can change (tenant view)

Depending on your deployment, tenant admins may be able to:

- override the prompt to match your business definitions
- tighten output requirements (e.g., "JSON only")
- update classification labels (dropdown values)
- enforce a scoring rubric

> Best practice: keep prompts *stable* once you start tracking metrics over time. Prompt changes can shift results and affect trend analysis.

What tenant admins can and cannot change

Can override:

- **Prompt** (Task instructions and Task inputs)
- **Filters** (eligibility criteria)

Cannot override (managed by provider):

- Attribute mapping (output fields)
- Response schema
- AI engine selection

This ensures consistent data structure across tenants while allowing customization of analysis logic and filtering rules.

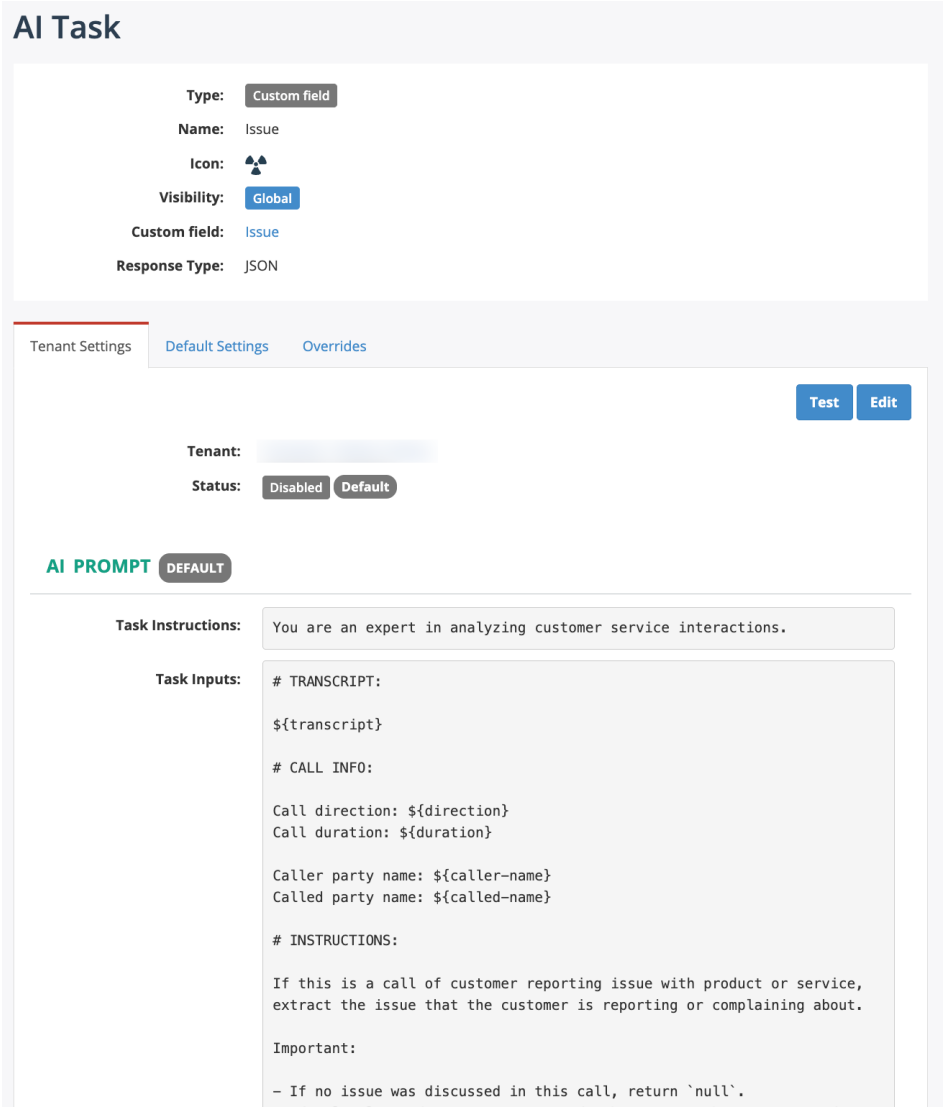

Figure: AI Task tenant view showing overridable settings.

4.4 Activate Prebuilt AI Tasks

Call Recording includes prebuilt AI Tasks (insights) that can be enabled per tenant. These tasks are often provided as “defaults” by Call Recording or your service provider, and you choose which ones to activate.

Menu path

`Administration > Speech Analytics > AI Assistant > AI Tasks`

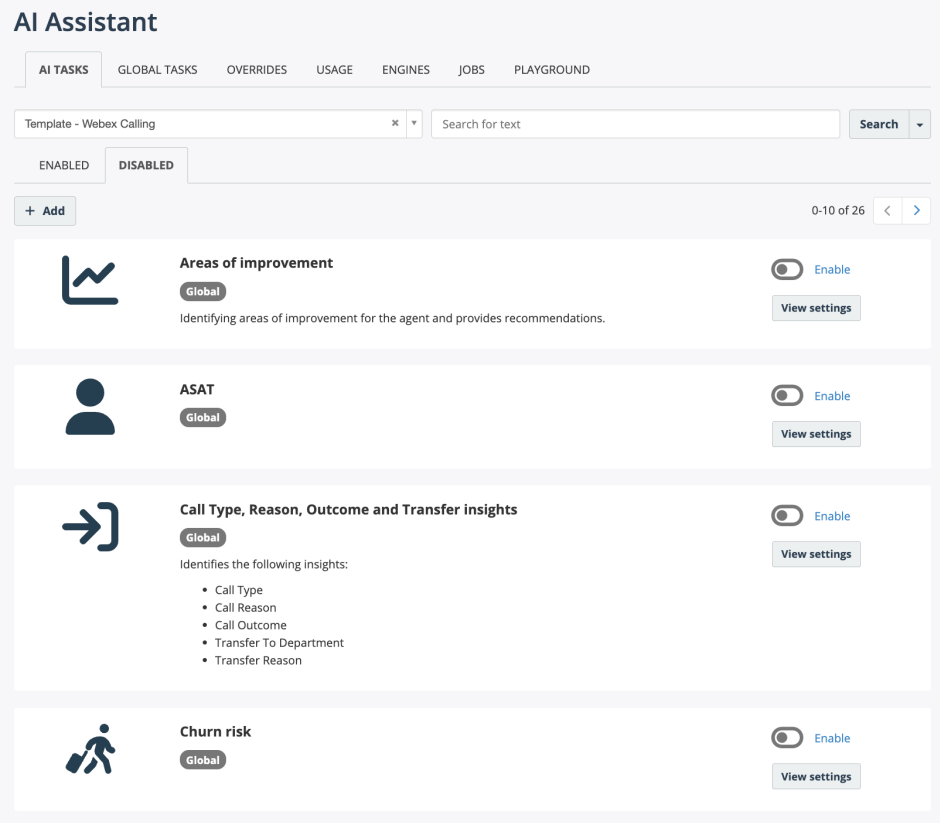

Figure: Disabled tab showing available prebuilt AI Tasks that can be enabled.

Enable a task for your tenant

1. Navigate to `Administration > Speech Analytics > AI Assistant > AI Tasks`.
2. Open the **Disabled** tab.
3. Find the AI Task you want to enable (e.g., “CSAT”, “Conversation Summarization”).
4. Click **Enable**.

Expected result

- The task moves to the **Enabled** tab.
- New eligible conversations (that match the task's filters) begin receiving outputs for the mapped Custom Fields.

Validate after enabling

After enabling a task:

1. Open a recent conversation that should qualify (has transcript/thread, matches filters).
2. Check **Conversation Details → Analytics** to see the populated fields and explanation.
3. Check **Dashboards** (if applicable) for newly populated metrics.
4. Use **Search** to filter by the new field (e.g., CSAT < 3).

Backfill behavior (historical conversations)

When you enable an AI Task:

- **New conversations** are processed automatically going forward
- **Historical conversations** are NOT automatically processed

If you need to process historical conversations, contact your Call Recording administrator or support team to request an on-demand backfill job. They can configure a one-time job to process conversations from a specific date range.

> **Note:** Backfill jobs may cause temporary spikes in dashboard metrics as historical data is populated.

Disabling a task

To stop populating a metric:

1. Go to the **Enabled** tab.
2. Find the task.
3. Click **Disable** (or equivalent).

What disabling does

- Stops future processing for that task (going forward)

- Existing Custom Field values are **preserved** (they remain visible in dashboards and conversation details)
- To clear existing values, contact your administrator

4.5 Override Prompt and Filters

Prebuilt AI Tasks often come with default prompts and filters. Call Recording allows tenant admins (where permitted) to override:

- **Prompt** (analysis instructions, scoring rubric, output format)
- **Filter** (which conversations qualify)
- or **both**

This enables customization without requiring a fully custom task.

Menu path

`Administration > Speech Analytics > AI Assistant > AI Tasks`

- Enable a task first (from the **Disabled** tab).
- Then open the task in the **Enabled** tab and click **Edit**.

When to override the prompt

Override the prompt when you need to:

- align definitions to your business (e.g., what counts as “resolved”)
- adjust scoring rubrics (CSAT, lead scoring, QA standards)
- enforce strict output rules (e.g., “JSON only”)
- include company-specific terminology (products, competitors)
- produce additional detail in the explanation (for coaching)

> Recommendation: avoid frequent prompt changes after rollout. Prompt changes can shift metrics and trend lines.

When to override filters

Override filters when you need to:

- exclude irrelevant conversation types (e.g., internal/test calls)
- require minimum duration or minimum content length
- restrict to specific channels (calls only) or teams/queues
- control cost by limiting scope

Step-by-step: override prompt and/or filter

1. Go to `Administration > Speech Analytics > AI Assistant > AI Tasks`.
2. Open **Enabled** and select the task.
3. Click **Edit**.
4. Choose what to override:
5. Prompt
6. Filter
7. Prompt + Filter
8. Make changes.
9. Click **Save**.
10. Immediately test the change using **Playground** or **Save and Test**.

Expected result

- The task continues to run for your tenant, but uses your overridden prompt and/or filter.

Guardrails (avoid breaking dashboards and mapping)

When overriding prompts:

- Keep the **output format** stable (especially JSON structure).
- Keep numeric ranges consistent (e.g., CSAT 1–5).
- Keep dropdown labels aligned to the Custom Field options.

If the AI returns an unexpected format: - values may not save into Custom Fields - dashboards and search filters may break or show blanks

Governance: tracking and change management

Recommended practices:

- Document why an override was made and when
- Test on a representative sample before broad rollout
- Avoid making changes during KPI reporting periods
- If you maintain multiple prompt versions, use version labels in the prompt header

Override visibility

Tasks with overrides display an "**Overridden settings**" tag in the task list, making it easy to identify which tasks have been customized.

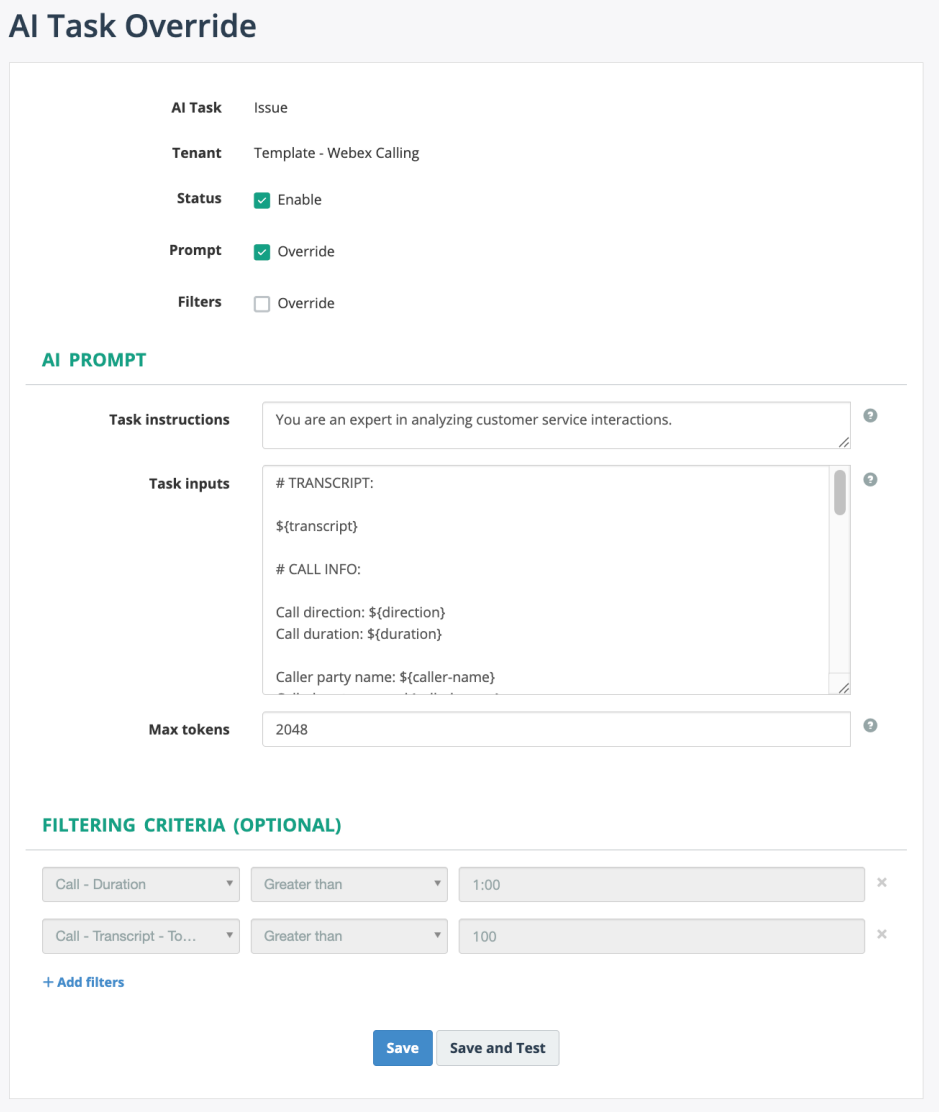

Figure: Override configuration showing checkboxes for Prompt and Filter overrides.

Override behavior with global updates

When your provider updates the global task definition:

- Your overridden settings (Prompt and/or Filter) **remain unchanged**
- Non-overridden settings receive the global update
- To adopt new global defaults, you can disable your override

Reverting to defaults

To revert an override to the global default:

1. Open the task and click **Edit**
2. Uncheck the **Override** checkbox for Prompt or Filter
3. Save the task

The task will then use the global default settings for that component.

4.6 Create Custom Insights (Tenant Tasks)

Custom insights let you extract business-specific data points from conversations—beyond the prebuilt catalog. Examples:

- Hospitality: reservation start date, total nights, VIP status
- Support: product area, issue category, escalation reason, resolution status
- Sales: urgency level, next actions, competitor mentioned, deal stage

A typical custom insight implementation consists of:

1. One or more **Custom Fields** to store the outputs
2. One **AI Task** that extracts the outputs and maps them to fields
3. Filters (optional) to scope which conversations are analyzed
4. Testing (Playground) and rollout

Step 1 — Define the insight

Write a clear definition before configuring anything:

- What question does the insight answer?
- What format should the value take?
- numeric / dropdown / date / free text
- What are the allowed values (if dropdown)?
- What does “unknown / not mentioned” mean?
- Do you need a human-readable **explanation**?

> Strong definitions lead to consistent extraction and better dashboards.

Step 2 — Create Custom Field(s)

Menu path: `Administration > Customization > Custom Fields`

Create the field(s) needed: - Value field (e.g., “VIP Status” dropdown) - Optional explanation field (if your deployment stores explanations in separate fields)

See: **Custom Fields for Insights**.

Step 3 — Create or clone an AI Task

Menu path: `Administration > Speech Analytics > AI Assistant > AI Tasks`

In most deployments, tenant admins cannot create new AI Tasks directly. Tasks are created at the platform level and made available to tenants. Tenant admins can:

- **Enable** prebuilt tasks from the Disabled tab
- **Override** prompt and filter settings on enabled tasks

If you need a custom task that doesn't exist in the catalog, contact your provider to request a new task definition. Once the task is available, you can enable it and customize the prompt/filters for your business needs.

Step 4 — Configure mapping (attributes → fields)

In the task configuration, map output attributes to Custom Fields.

Example (Hospitality): - `reservation_start_date` → “Reservation Start Date” (Date) - `reservation_total_nights` → “Reservation Total Nights” (Number) - `vip_status` → “VIP Status” (Dropdown)

If storing explanations as separate fields: - `vip_status_explanation` → “VIP Status Explanation” (Text)

Step 5 — Write a prompt that is strict and testable

Recommended prompt pattern:

- tell the AI what to extract
- define allowed values/ranges
- instruct “JSON only”
- include an “unknown” rule
- include an explanation

Example (VIP Status)

```
TRANSCRIPT:
${transcript}

TASK:
Determine if the customer is a VIP based only on the transcript.
```

```
OUTPUT RULES:
- vip_status must be one of: "VIP", "Non-VIP", "Unknown"
- Use "Unknown" if the transcript does not contain enough evidence.
- Provide a short explanation (1–2 sentences) referencing evidence from the transcript.
- Output must be valid JSON only.

RESPONSE JSON:
{
  "vip_status": { "value": "VIP", "explanation": "..." }
}
```

Step 6 — Add filters (optional but recommended)

Use filters to avoid irrelevant conversations, for example: - only inbound support calls - only conversations longer than 15 seconds - only a specific channel (calls vs chat)

Step 7 — Test in Playground and validate

Before rollout: - test on 5–10 representative conversations - confirm JSON is valid - confirm values match allowed types/ranges - confirm explanations are useful and concise

Step 8 — Roll out dashboards and saved searches

- Create dashboards for key numeric/dropdown fields
- Create saved searches for operational triage:
- “VIP = VIP”
- “Reservation Start Date within next 7 days”
- “Urgency = High”

4.7 Auto QA (Scorecards and Tasks)

Auto QA evaluates conversations against a structured **QA Scorecard** (sections and questions). Unlike simple metrics (CSAT, sentiment), Auto QA typically requires configuring:

1. A **Scorecard** (your QA rubric)
2. An **Auto QA AI Task** that applies the scorecard to eligible conversations
3. Reporting/dashboards to view scores and trends

What Auto QA is best for

- consistent evaluation across large conversation volumes
- coaching and training: identify question-level gaps
- compliance: detect missing required statements
- operational review: prioritize low-scoring conversations

Step 1 — Define the QA Scorecard

A scorecard usually includes: - **sections** (e.g., Greeting, Discovery, Resolution, Compliance) - **questions** (binary/graded) - **scoring weights** (optional) - **pass/fail thresholds** (optional)

Recommended design principles

- Keep question wording objective and evidence-based.
- Avoid compound questions (split into separate items).
- Prefer measurable behaviors (“Did the agent confirm identity?”) over vague ones (“Was the agent professional?”).
- Include a “Not applicable” option if relevant.

Step 2 — Configure the scorecard in Call Recording

QA Form configuration

Configure your QA scorecard with sections and questions:

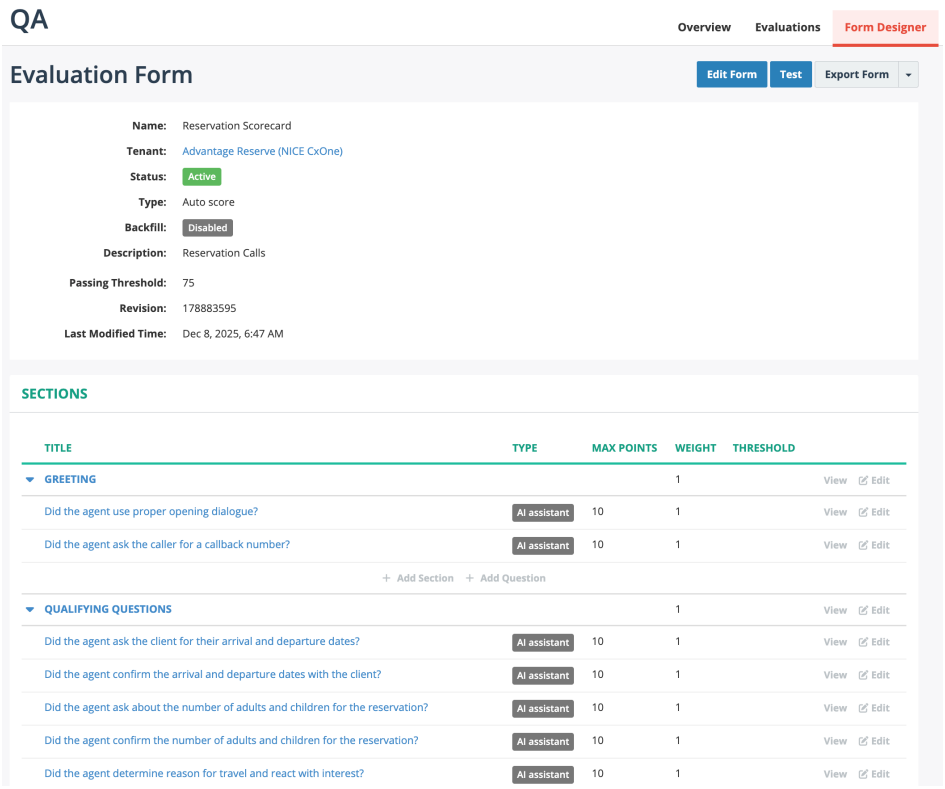

Figure: Auto QA form showing sections and questions in the scorecard.

Question configuration

Each question can be configured with:

- Question text
- Answer type (Yes/No, scale, dropdown)
- Scoring weight
- N/A option if applicable

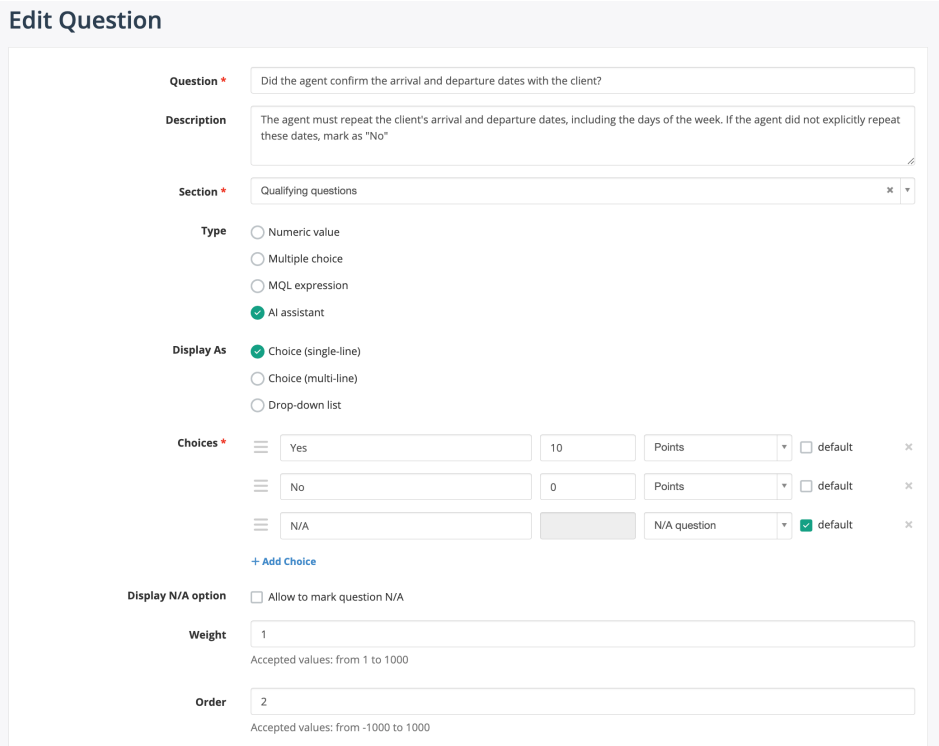

Figure: Question configuration with answer options and scoring.

Auto QA results

Auto QA results are displayed in:

- **Dedicated QA view** on the conversation details page (detailed per-question breakdown)
- **Dashboards** for high-level score trends and distributions

Step 3 — Enable or configure the Auto QA AI Task

Auto QA requires an AI Task that: - uses the scorecard as evaluation criteria - outputs per-question results and an overall score - includes a short explanation/rationale per question (recommended)

If Auto QA is provided as a prebuilt task: - enable it from `AI Tasks` (Disabled → Enable) - configure filters (e.g., only certain queues)

If it requires a tenant-specific configuration: - create the task and connect it to the scorecard (deployment-dependent)

Step 4 — Test and calibrate

Use Playground or “Save and Test”: - run Auto QA on a sample set - compare results to human QA - refine question wording and scoring thresholds

Step 5 — Reporting and coaching workflow

Define how results are used: - dashboards: overall QA score trends, distribution buckets, low-score drilldowns - saved searches: conversations with “Compliance: Failed” - coaching: question-level remediation and training

4.8 Permissions and Visibility

To roll out Conversation Analytics successfully, ensure the right people can:

- view transcripts/threads
- view AI insights and explanations
- access dashboards and saved searches
- (for admins) enable/override AI Tasks and manage Custom Fields

Recommended role model (typical)

A common approach:

- **Tenant Admin**: full configuration access (fields, tasks, overrides, dashboards)
- **Supervisor / QA Lead**: dashboards, advanced search, conversation details (read/write coaching notes if applicable)
- **Analyst**: dashboards and exports, limited configuration
- **Agent / User**: view assigned conversations, limited analytics

Field visibility

If your deployment supports field-level visibility: - keep sensitive insights restricted (e.g., churn risk, compliance flags, revenue metrics) - make operational insights broadly visible (e.g., topics, summaries)

Call Recording uses role-based permissions to control access to features. The specific roles and their permissions may vary by deployment.

> Contact your provider if you need to adjust role permissions or restrict access to specific fields or dashboards.

Governance: who can change AI behavior

Prompt/filter overrides can change metrics and workflows. Decide:

- Who can enable/disable tasks?
- Who can override prompts/filters?
- Who can create custom insights?

> Recommendation: limit prompt overrides to a small admin group and require testing before rollout.

5. Testing and Tuning

5.1 Playground

Playground helps admins test and refine AI Tasks against real conversations before rolling changes out widely.

Playground is especially useful when: - tuning prompts for consistency - validating output schema (JSON) - checking edge cases (angry customers, partial resolution, transfers) - confirming explanations are useful for human review

How to access testing

To test an AI Task, open the task view page and click the **Test** button.

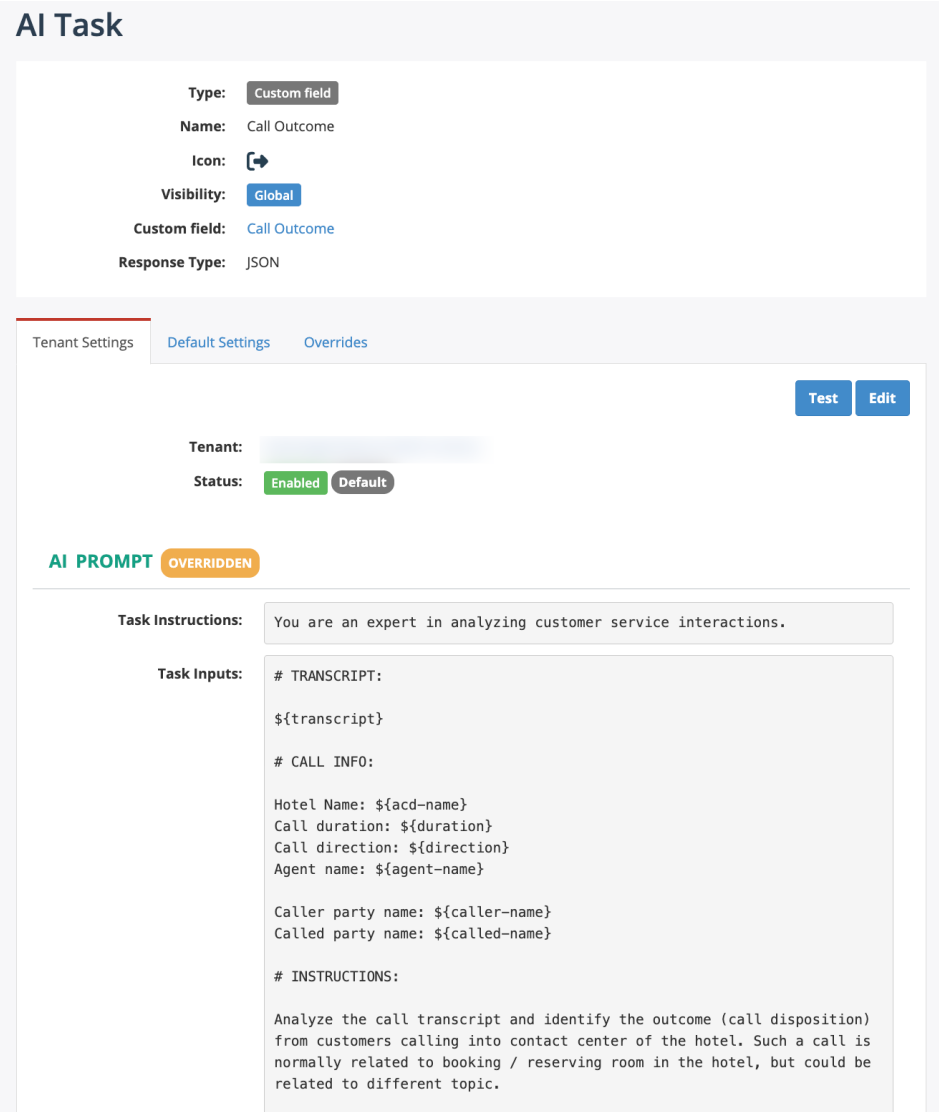

Figure: AI Task view showing the Test button to access prompt testing.

Testing workflow

Step 1 — Select a call

Choose a conversation to test against:

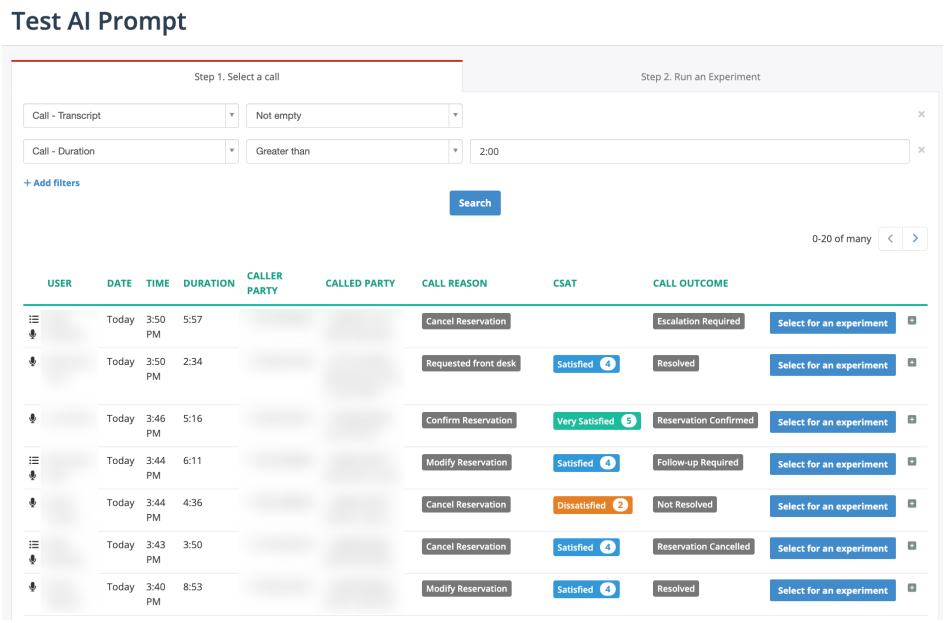

Figure: Select a call from the list to use for testing.

Step 2 — Run an experiment

View the transcript and run the prompt against it:

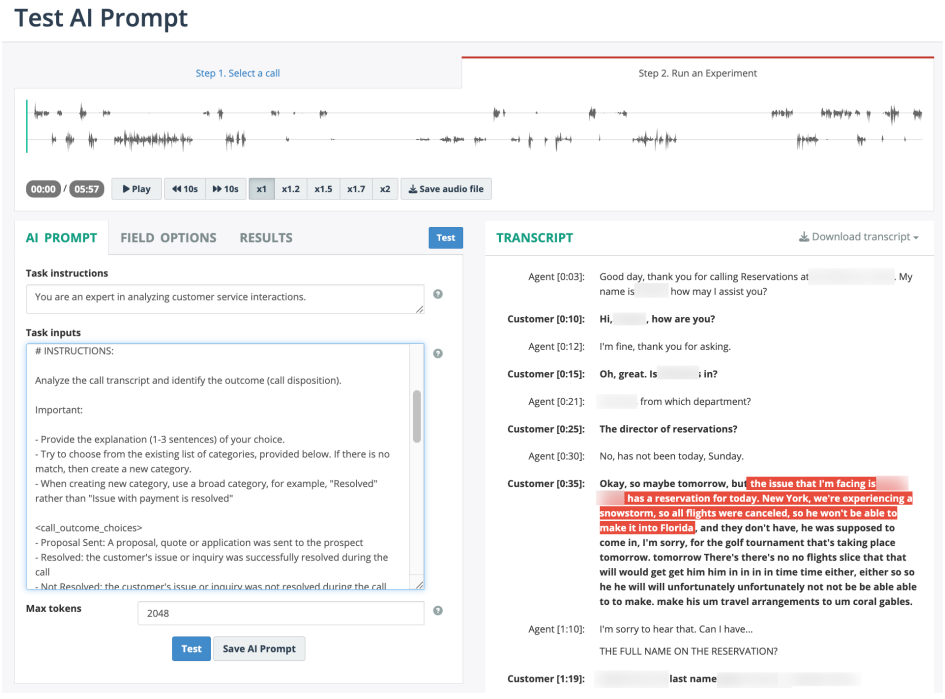

Figure: Run Experiment view showing the transcript and prompt configuration.

Step 3 — View results

Inspect the AI output to verify correctness:

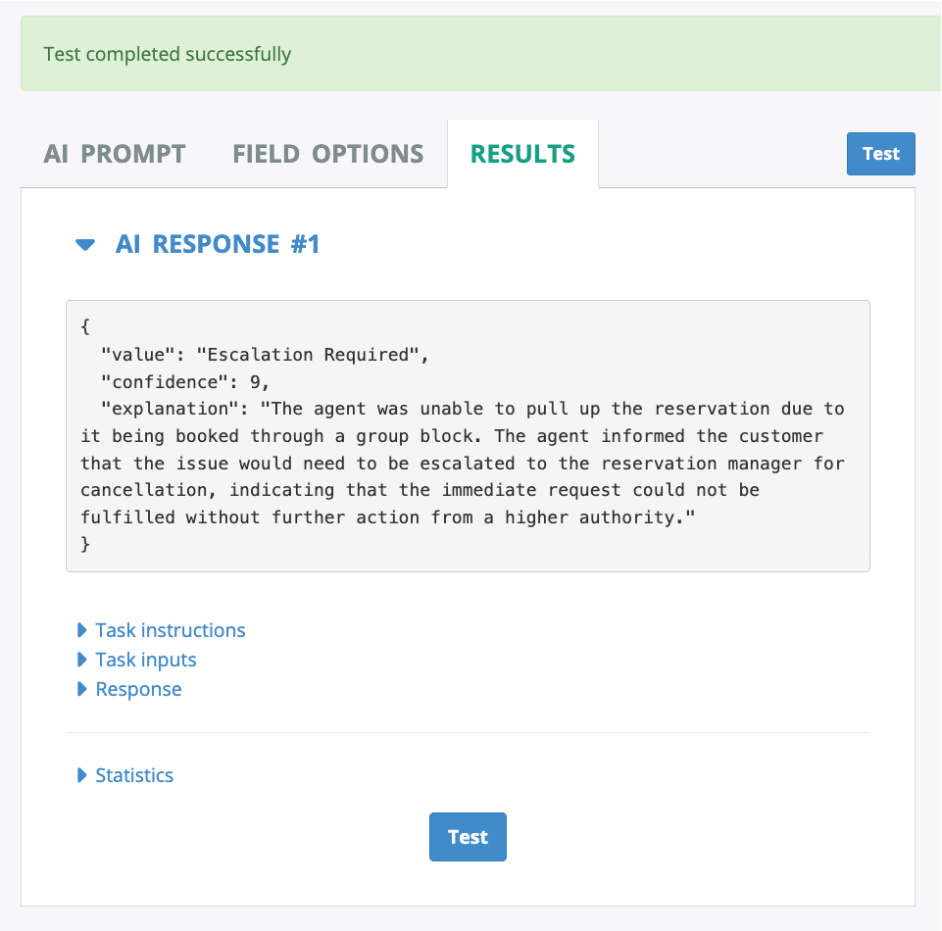

Figure: Experiment results showing the AI-generated output.

If the prompt produces good results, click **Save AI Prompt** to save your changes.

Recommended testing workflow

Step 1 — Choose representative conversations

Pick 5–10 conversations that represent: - typical cases - edge cases - different agents/queues - different customer sentiment levels

Step 2 — Run a single AI Task

- select the task to test (e.g., CSAT Scoring)
- run it against one conversation transcript/thread

Step 3 — Inspect output

Verify: - output is valid JSON (if configured as JSON) - values are in the expected range (e.g., CSAT 1–5) - dropdown values match allowed labels - explanation is concise and evidence-based

Step 4 — Iterate safely

- adjust prompt wording (clarify rubric, define “unknown”, tighten output)
- adjust filters if too many irrelevant calls qualify
- re-run on the same conversation and compare results

Step 5 — Validate before rollout

Before enabling a task broadly or changing a prompt: - validate on at least 10 conversations - confirm acceptable consistency for your stakeholders

Tips for faster calibration

- Add 2–3 “anchor examples” in your validation set (known good/bad calls)
- Use consistent definitions (write them in a shared document for stakeholders)
- Keep explanations short (1–3 sentences)

5.2 Prompting Guidelines

This chapter provides practical guidance for writing and overriding AI Task prompts in Call Recording.

Use strict, structured output for metrics

For any insight used in dashboards, search, or automation, prefer **JSON** output with a stable structure.

Recommended pattern:

- return **value**
- return **explanation** (1–3 sentences)

Example:

```
{
  "csat": {
    "value": 4,
    "explanation": "The agent resolved the issue and the customer expressed thanks and satisfaction."
  }
}
```

Make allowed values explicit

Numeric scores

- define the range (e.g., 1–5)
- define what each score means
- define what to do when evidence is insufficient

Dropdown classifications

- list allowed labels exactly
- instruct the model to pick only from that list
- include “Unknown” when appropriate

Example:

```
Allowed values for churn_risk: "Low", "Medium", "High", "Unknown"
Return "Unknown" if the transcript does not provide enough evidence.
```

Be explicit about evidence and explanations

Explanations are most useful when they: - reference specific statements or moments in the transcript/thread - avoid speculation - remain concise

Good instruction:

```
Write a 1–2 sentence explanation that cites evidence from the transcript.
Do not invent details not present in the conversation.
```

Enforce “JSON only”

A common failure mode is extra text before/after JSON. Add rules such as:

- “Output **must be valid JSON only**.”
- “Do not include Markdown.”
- “Do not include commentary outside the JSON.”

Handle ambiguity (Unknown / Not mentioned)

Define an explicit “unknown” policy: - Use “Unknown” if evidence is insufficient - Use null/empty string only if your schema allows it - Avoid guessing

Keep prompts stable after rollout

Prompt changes can shift results and trend lines. Recommended approach: - test in Playground - announce prompt changes to stakeholders - keep a prompt version header inside the prompt text

Example:

```
Prompt version: CSAT-v3 (2026-01-24)
Change note: clarified what counts as "resolved"
```

Suggested prompt structure

Many teams keep prompts consistent across tasks using this structure:

1. Role / expertise
2. Task description
3. Definitions/rubric
4. Rules (evidence-based, unknown handling)
5. Output schema (JSON)

Multi-output tasks (one task writes multiple fields)

When a task writes multiple fields: - include a single JSON object with multiple keys - keep key names aligned to mapping - ensure every key has stable types

Example:

```
{
  "churn_risk": { "value": "High", "explanation": "..." },
  "escalation_reason": { "value": "Billing dispute", "explanation": "..." }
}
```

Common pitfalls and how to avoid them

- **Pitfall:** vague rubrics (“good/bad”)
  Fix: define explicit criteria and examples.
- **Pitfall:** free-text categories (“customer was unhappy about shipping delays”)
  Fix: force dropdown values.
- **Pitfall:** too-long explanations
  Fix: cap at 1–3 sentences and require transcript evidence.
- **Pitfall:** mixing multiple unrelated insights in one task
  Fix: group only tightly related outputs.

5.3 Validation Checklist

Use this checklist before enabling a new AI Task, changing a prompt, or rolling out a new custom insight.

A. Data prerequisites

- Conversations have transcripts/threads available
- Conversation volume is sufficient to validate (recommended: 10–30 samples)
- Conversations represent your typical and edge cases

B. Task configuration checks

- Task is mapped to the intended Custom Field(s)
- Field types match output types (number/date/dropdown/text)
- Filters are appropriate (not overly strict, not overly broad)

C. Output format checks (JSON recommended)

- Output is valid JSON (no extra text outside JSON)
- Keys match mapping attributes exactly
- Values are within allowed ranges / labels
- “Unknown” or “not mentioned” behavior is defined and consistent

D. Explanation quality checks

- Explanations are concise (1–3 sentences)
- Explanations cite evidence from transcript/thread
- Explanations avoid speculation (“maybe”, “probably”) unless explicitly allowed

E. UI verification

After enabling/adjusting the task:

- Field values appear in **Conversation Details → Analytics**
- Dashboards update correctly (if applicable)
- Search filters work (e.g., numeric comparisons like CSAT < 3)
- Drilldowns work (bucket labels navigate to matching conversations)

F. Governance checks

- Stakeholders agree with definitions (QA/CX/Sales)
- Prompt version/change notes are documented
- Rollout plan defined (pilot group first, then broader)

6. Dashboards and Search

6.1 Dashboards for Metrics

Dashboards help teams monitor trends and drill down into conversations that need attention. In Call Recording, dashboards are typically powered by **Custom Fields** populated by AI Tasks.

Common dashboard patterns

- **Distribution buckets** (e.g., CSAT: Very Dissatisfied → Very Satisfied)
  Useful for quickly finding “bad” conversations.
- **Trend over time** (e.g., average CSAT daily/weekly)
- **Breakdowns** by team/queue/agent/channel (depending on permissions and metadata)

Creating dashboards from Custom Fields

Some Custom Fields can automatically create a dashboard when enabled (e.g., “Create a dashboard for this metric”).

Menu path for the field: `Administration > Customization > Custom Fields`

Recommended settings (for numeric metrics): - aggregation: **Average** - thresholds: configured for bucket labels and colors

Expected result

The dashboard displays: - average value - distribution buckets (clickable) - trend chart

Clickable buckets (drilldowns)

When thresholds are configured, users can click a bucket (e.g., “Dissatisfied”) to navigate to a conversation list filtered by that bucket.

This enables workflows such as: - review all CSAT ≤ 2 calls - identify top drivers using explanations - assign coaching actions

Dashboard governance tips

- Keep definitions stable (avoid frequent prompt changes) or document changes clearly.
- Use consistent thresholds across teams.
- Create "operational dashboards" (daily triage) and "executive dashboards" (weekly trends).

Dashboard access

Dashboards are accessible from the **Speech Analytics** section of the main navigation. Dashboard visibility is controlled by user roles and permissions configured by your administrator.

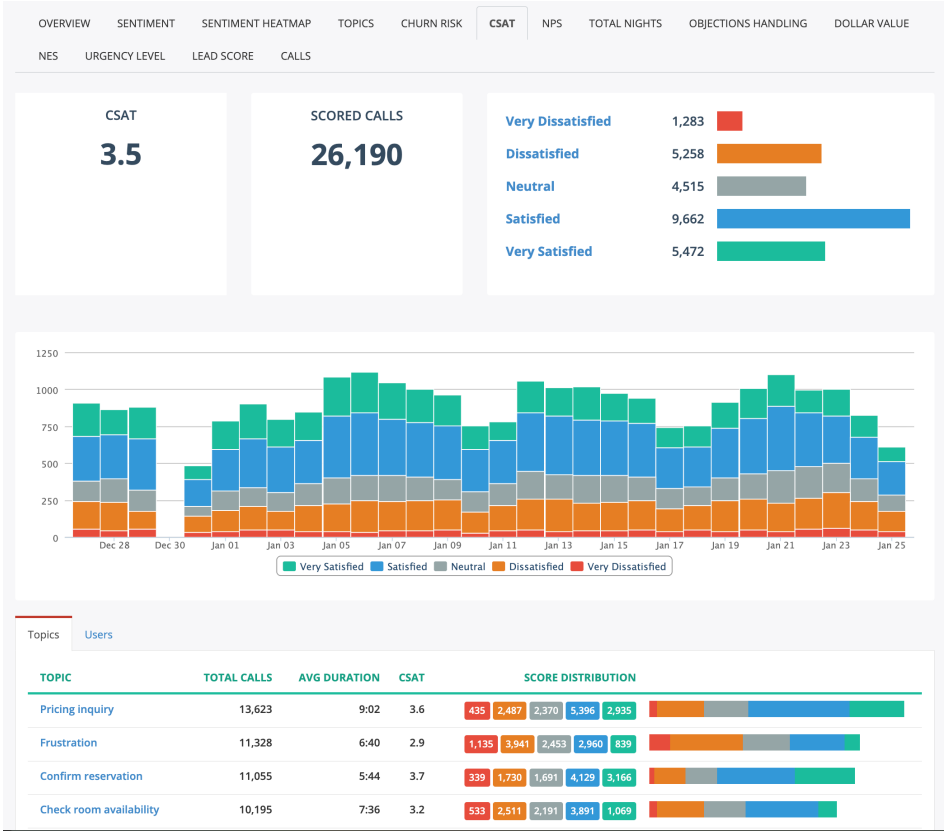

Figure: CSAT Dashboard showing score distribution, trend over time, and per-user breakdown.

Dashboard components

Each metric dashboard typically includes:

- **Summary tiles** – Total count and average/aggregate value
- **Distribution chart** – Breakdown by threshold buckets (clickable for drilldown)
- **Trend chart** – How the metric changes over time
- **Per-user table** – Breakdown by agent for coaching and comparison

6.2 Search and Saved Views

Search lets supervisors and analysts quickly find conversations that match AI insight criteria (e.g., “CSAT < 3”, “Churn Risk = High”, “Competitor mentioned”).

Searching by AI insight values

Once AI Tasks populate Custom Fields, those fields become available as filters in search.

Example: - Find dissatisfied conversations by CSAT: - Field: **CSAT** - Condition: **Less than** - Value: **3**

> Tip: Use thresholds + dashboards for discovery, then search filters for precise targeting.

Combining filters

Common combinations: - CSAT < 3 AND Topic = “Billing” - Churn Risk = High AND Escalation Reason is not empty - Competitor Mentioned = Yes AND Lead Stage = Late

Saved views (recommended for operations)

Saved views allow teams to: - monitor recurring queues (low CSAT, high churn risk) - standardize coaching workflows - reduce manual filter setup

Suggested saved views

- “Low CSAT (≤ 2) – Last 7 days”
- “High Churn Risk – This week”
- “Escalations – Top 3 reasons”
- “Sales Objections – Competitor mentions”

Access search via the **Calls** or **Conversations** list. A filter panel on the left side allows you to add criteria for AI insight values, call metadata (direction, duration, agent, queue), and date ranges.

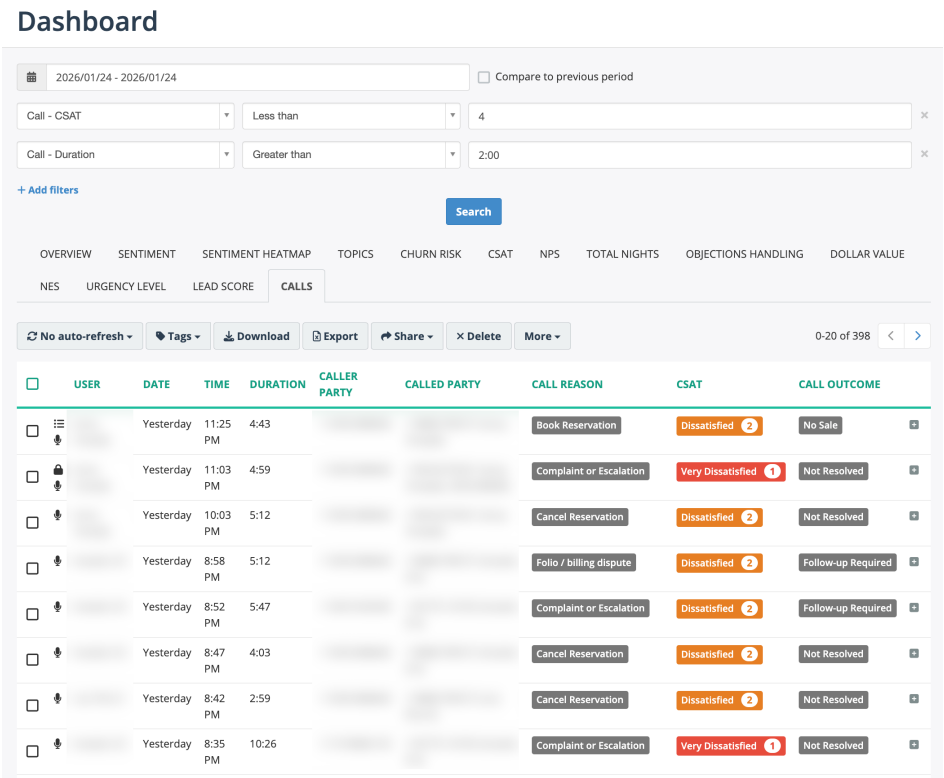

Figure: Conversation list with search filters applied.

6.3 Reports and Exports

Reports and exports help you distribute insights beyond Call Recording dashboards—for example to BI tools, spreadsheets, or operational reviews.

Common export use cases

- weekly CX summary: CSAT distribution + top issues
- QA tracking: low QA score conversations for coaching
- sales coaching: top objections and next actions
- operational triage: export “CSAT < 3” list for follow-up

What to export

Most teams export: - conversation metadata (date, agent, queue, channel) - transcript/thread (if permitted) - key Custom Fields (CSAT, sentiment, topics, churn risk) - explanations (if available)

Export options vary by deployment. Common capabilities include:

- **CSV export** of conversation lists with selected fields
- **Custom Fields** included in exports (CSAT, sentiment, topics, etc.)
- **Transcript export** may be available depending on permissions

Contact your provider for specific export capabilities and scheduled report options.

7. Monitoring and Troubleshooting

7.1 Admin Monitoring (Tenant View)

This chapter describes what tenant admins can monitor to ensure Conversation Analytics is working as expected.

What to monitor regularly

- **Transcription health** (voice calls)
- transcripts present for recent calls
- language is correct
- **AI Task enablement**
- required tasks are enabled for the tenant
- filters are not excluding everything
- **Output freshness**
- recent conversations show populated fields
- dashboards update with expected latency
- **Override tracking**
- know which tasks have custom prompt/filter overrides
- document and review changes periodically

Where to monitor

AI Tasks status

`Administration > Speech Analytics > AI Assistant > AI Tasks`

- Check **Enabled** tasks
- Review filters and prompt overrides (if any)

Conversation spot-checking

- Pick a handful of recent conversations and confirm:

- transcript/thread exists
- insights show under Conversation Details → Analytics
- explanations look reasonable

Dashboards trend checks

- Watch for sudden discontinuities that may indicate:
- prompt changes
- filter changes
- ingestion/transcription changes

Overrides visibility

Tasks with tenant-level overrides display an "Overridden settings" tag on the task card.
Review these periodically to track what has been customized.

> Note: Job status, engine details, and usage metrics are managed at the platform level and are not typically visible to tenant admins. If you need this information, contact your provider.

What to collect for support (when something is wrong)

When contacting your provider support, collect:

- tenant name / ID
- time range of affected conversations
- example conversation IDs
- which AI Task(s) are missing or wrong
- whether the tasks are enabled and any overrides were made
- screenshots of:
- conversation analytics view
- task configuration (prompt/filter)

7.2 Troubleshooting

This chapter provides tenant-facing troubleshooting steps for common issues.

Symptom: Insights are not appearing on conversations

Checklist

1. Does the conversation have a transcript/thread?
2. For calls, confirm transcription is enabled and a transcript exists.
3. Is the AI Task enabled?
4. `Administration > Speech Analytics > AI Assistant > AI Tasks` → **Enabled**
5. Does the conversation match the task's filters?
6. duration threshold
7. direction (inbound/outbound)
8. channel restrictions
9. Is there a processing delay?
10. newly ingested conversations may take time to populate
11. Did you recently enable the task?
12. confirm whether historical backfill is supported (deployment-specific)

Symptom: Some conversations have insights, others do not

Likely causes: - filters exclude certain conversation types - transcription missing for some calls - conversations too short - channel-specific availability (text channels not enabled yet)

Symptom: Values are blank or dashboards show "No data"

Checklist: - confirm Custom Field exists and is enabled - confirm field is "available to AI Insights tasks" - confirm the AI Task mapping points to the correct field - confirm output format matches the expected type (number vs dropdown vs date)

Symptom: Values look wrong or inconsistent

Checklist: - review the prompt definition and rubric - validate “unknown/insufficient evidence” policy - test the task in Playground on a known set of conversations - check whether the prompt was recently overridden or updated - confirm dropdown labels exactly match allowed values

Symptom: Explanations are missing or not helpful

Possible causes: - prompt does not request an explanation - explanation is not stored/displayed in your deployment - prompt requests long, generic text

Fix: - update prompt to require explanation (1–3 sentences) and evidence from transcript - if explanations require a separate field, create and map it

Call Recording stores explanations automatically alongside insight values. The explanation is displayed in the Conversation Details → Analytics view next to the insight value without requiring a separate custom field.

Symptom: Task enabled but still not running

Possible causes: - AI engine/provider issue (provider-managed in SaaS) - task execution backlog - temporary outages

What tenant admin can do: - capture example conversation IDs - confirm enablement + filters - contact provider with evidence

7.3 FAQ

Do AI insights work without transcription?

For **voice calls**, transcription is required because the AI analyzes the transcript. For **text channels** (chat/email/tickets), Call Recording analyzes the message thread directly (as enabled in your deployment).

Why don't I see the option described in this guide?

Some settings may be role specific or managed by the Service Provider. If you don't see a menu or option, contact your Call Recording Admin who may in turn contact the support team.

If I change a prompt, will historical metrics change?

It depends on whether your deployment supports backfill/reprocessing. Even without backfill, prompt changes affect future results and may shift trends.

Can one AI Task populate multiple fields?

Yes. An AI Task can map multiple output attributes to multiple Custom Fields, allowing you to extract several related insights in one run.

How should we handle "unknown" cases?

Define an explicit "Unknown" policy in the prompt and schema (recommended). This avoids the model guessing and improves trust.

8. Insight Types

8.1 Customer Experience Metrics

CSAT (Customer Satisfaction Score)

CSAT is an AI insight that estimates how satisfied the customer is with the outcome of a conversation. In Call Recording, CSAT is typically represented as:

- a numeric score (commonly **1–5**)
- optional thresholds/labels (e.g., “Dissatisfied”)
- a short explanation to help human reviewers trust and act on the score

Where CSAT appears

Once enabled and populated, CSAT is available in:

- **Conversation Details → Analytics** (score + label + explanation)
- **Dashboards** (average, distribution buckets, clickable drilldowns)
- **Search** (e.g., CSAT < 3)

What gets configured

Depending on your deployment, CSAT is configured using:

1. **Custom Field** (numeric) – stores the CSAT value and powers dashboards/search
   Menu: `Administration > Customization > Custom Fields`
2. **AI Task** (CSAT scoring) – extracts CSAT from transcript/thread and writes to the field
   Menu: `Administration > Speech Analytics > AI Assistant > AI Tasks`
3. Optional **Prompt/Filter override** – to align scoring to your business definitions

> In many deployments, CSAT is provided as a prebuilt AI Task and may already have a prebuilt Custom Field. In that case, you primarily **enable the task** and validate results.

Step 1 — Create or verify the CSAT Custom Field

Menu: `Administration > Customization > Custom Fields`

RECOMMENDED FIELD SETTINGS (CLASSIC CSAT)

- **Name:** CSAT
- **Type:** Number (integer)
- **Range:** 1–5 (optional: allow 0 = Unknown)
- **Display group:** CX Metrics (or similar)
- **AI Insights eligibility:** enabled (so tasks can map to this field)
- **Dashboard:** enabled (recommended)
- aggregate: Average

THRESHOLDS (LABELS + COLORS)

Recommended thresholds:

Score	Label
1	Very Dissatisfied
2	Dissatisfied
3	Neutral
4	Satisfied
5	Very Satisfied

> These thresholds create the dashboard buckets that users can click to drill down into matching conversations.

IF YOU DON'T SEE CSAT IN CUSTOM FIELDS

- In SaaS deployments, fields may be pre-configured by Call Recording.
- Contact your provider to enable the CSAT field for your tenant, or ask for permission to manage Custom Fields.

Step 2 — Enable the CSAT AI Task

Menu: `Administration > Speech Analytics > AI Assistant > AI Tasks`

1. Open the **Disabled** tab.
2. Find the CSAT task (name may vary: “CSAT”, “CSAT Scoring”).
3. Click **Enable**.

VALIDATE MAPPING

Open the enabled task and verify the mapping includes: - output attribute (e.g., `csat`) → Custom Field (CSAT)

Step 3 — (Optional) Override CSAT prompt and/or filters

Menu: `Administration > Speech Analytics > AI Assistant > AI Tasks` → **Enabled** → task → **Edit**

WHEN TO OVERRIDE

Override the prompt when: - you want stricter definitions of “resolved” - you want to weigh tone vs outcome differently - you want more consistent “unknown” handling - you need a more helpful explanation for coaching

Override filters when: - you want to score only certain call types (e.g., inbound support calls) - you want a minimum duration threshold (e.g., > 15 seconds)

RECOMMENDED FILTERS

- Call duration > 15 seconds (or > 30 seconds if you have many short calls)
- Inbound calls only (if CSAT is support-focused)

Step 4 — Recommended CSAT prompt pattern (value + explanation)

Call Recording prompts commonly extract: - the CSAT score - plus an explanation

SUGGESTED JSON FORMAT

```
{
  "csat": {
    "value": 4,
    "explanation": "The issue was resolved and the customer expressed appreciation before ending the
call."
```

```
  }
}
```

SUGGESTED RUBRIC (1–5)

- 1 = Very Dissatisfied: angry/frustrated, unresolved, negative outcome
- 2 = Dissatisfied: mostly negative, unresolved or poor handling
- 3 = Neutral: mixed/unclear, partial resolution
- 4 = Satisfied: resolved, generally positive
- 5 = Very Satisfied: praise/thanks, excellent outcome

Call Recording stores explanations automatically alongside the insight value. The explanation is displayed in the Conversation Details → Analytics view next to the CSAT score without requiring a separate custom field.

Step 5 — Test and validate

Use Playground or “Save and Test”:

1. Test on at least 10 conversations:
2. 3 clearly positive
3. 3 clearly negative
4. 4 mixed/ambiguous
5. Confirm:
6. output is valid JSON (if configured)
7. values are within 1–5
8. explanations are short and evidence-based

Step 6 — Verify dashboards and search

DASHBOARDS

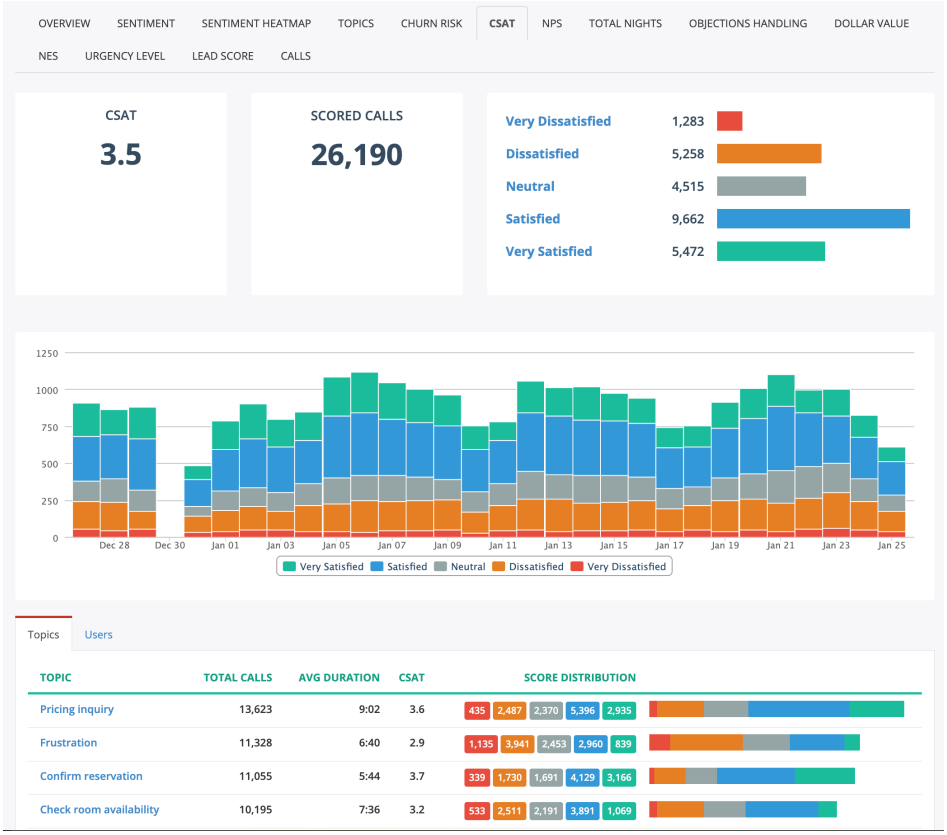

Figure: CSAT Dashboard showing score distribution, trend over time, and per-user breakdown.

- Open the CSAT dashboard
- Confirm bucket counts and clickable drilldowns work as expected

SEARCH

Use advanced search to filter: - CSAT < 3 (dissatisfied calls) - CSAT = 1 (urgent review)

Troubleshooting

CSAT not showing up - transcript missing - task not enabled - filters exclude the conversation - mapping points to a different field

Dashboard empty - metric field exists but isn't being populated yet - time range too narrow - no backfill of historical data

NPS (Net Promoter Score)

NPS is traditionally based on a survey question (“How likely are you to recommend us to a friend or colleague?” on a 0–10 scale). In Conversation Analytics, NPS is often implemented as an **AI-estimated proxy** based on the conversation content.

> Recommendation: be explicit with stakeholders that this is **AI-estimated NPS**, unless you are extracting an explicitly stated rating from the conversation.

Common NPS implementation options

Choose one approach and document it internally to avoid confusion:

1. **Extract explicit NPS rating**
   Use when agents ask the NPS question and customers answer with a number.
2. **Infer NPS from sentiment + outcome** (proxy)
   Use when no explicit question is asked; AI estimates likely rating.
3. **Hybrid**
   Extract explicit rating when present; otherwise return “Unknown”.

Recommendation: Use **Hybrid** approach—extract explicit ratings when customers state them, otherwise return "Unknown". This avoids inventing numbers when NPS is not discussed in the conversation.

Configuration overview (tenant view)

- Create/verify a numeric Custom Field (0–10) or dropdown (Promoter/Passive/Detractor)
- Enable the prebuilt NPS AI Task (if available) or create a tenant task
- Configure thresholds/buckets:
- 0–6 = Detractor
- 7–8 = Passive
- 9–10 = Promoter
- Test and validate on representative conversations

Where NPS appears

- Conversation Details → Analytics
- Dashboards (distribution by bucket)

- Search (e.g., NPS <= 6)

Prompt recommendations

- Prefer JSON output
- Include “Unknown” when evidence is insufficient
- Provide a short explanation

NES (Net Easy Score)

NES (sometimes called Customer Effort Score) measures how easy it was for the customer to get help or resolve an issue. In Conversation Analytics, NES can be:

- extracted explicitly (if asked in the conversation), or
- inferred as a proxy from friction, effort, transfers, and resolution quality

Configuration overview (tenant view)

1. Create/verify an NES Custom Field (commonly 1–5 or 1–7 depending on your standard).
2. Enable the NES AI Task (prebuilt) or create a tenant task.
3. Configure thresholds/labels (e.g., Easy ↔ Difficult).
4. Test and validate.

Recommendation: Use a **1–5 scale** for simplicity, unless your organization already uses a different standard (1–7 or 0–10). You can also use a dropdown (Easy/Neutral/Difficult) for clearer reporting.

Prompt recommendations

- Define the scale explicitly
- Instruct to weigh:
- number of transfers/repeats
- clarity of resolution
- customer confusion/frustration
- Require a short explanation with evidence

Churn Risk

Churn Risk estimates the likelihood that the customer may churn (cancel, switch providers, reduce usage) based on the conversation content.

Recommended representation

- **Dropdown**: Low / Medium / High / Unknown
- Optional supporting fields:
- Churn Risk Reason (text)
- Evidence snippets (text)

Configuration overview (tenant view)

1. Create/verify Custom Field(s)
2. Churn Risk (dropdown)
3. Optional: Churn Risk Reason (text)
4. Enable the prebuilt Churn Risk AI Task (or create a tenant task).
5. Configure filters:
6. focus on retention-sensitive conversations (support cancellations, billing disputes)
7. Test and calibrate with stakeholders.

Prompt recommendations

- Define what signals “High” vs “Medium” vs “Low”
- explicit cancellation intent
- competitor mention
- repeated unresolved issues
- strong negative sentiment + unresolved outcome
- Require “Unknown” when insufficient evidence
- Require explanation referencing transcript evidence

EXAMPLE JSON FORMAT

```
{
  "churn_risk": { "value": "High", "explanation":
"Customer threatened to cancel due to repeated billing issues." },
```

```
  "churn_risk_reason": { "value": "Billing issues and cancellation intent", "explanation": "Customer
said they will cancel if billing is not fixed." }
}
```


Common use cases

- **Supervisor triage:** Review high-risk conversations for immediate follow-up
- **Retention team workflow:** Route high-risk customers to retention specialists
- **Executive dashboards:** Track churn risk trends as a leading indicator

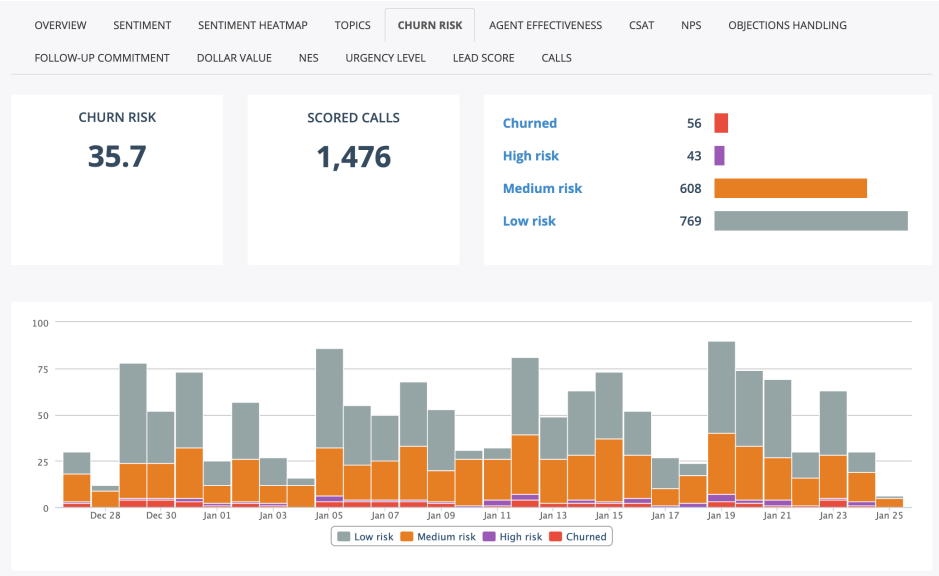

Figure: Churn Risk dashboard showing distribution and trend.

Top Issues Reported

"Top Issues Reported" categorizes what customers are contacting you about (e.g., Billing, Login issues, Shipping delays). This insight is most useful when it uses a **controlled taxonomy** (dropdown labels) rather than free text.

Recommended representation

Choose one:

- **Single dropdown** (one primary issue)
- **Multi-select dropdown** (multiple issues per conversation)
- **Hybrid**: primary issue (dropdown) + issue details (text)

Recommendation: Start with **single primary issue** (dropdown) to keep dashboards clean and actionable. Add an "Issue Details" text field if you need to capture additional context.

Configuration overview

1. Define the taxonomy (10–30 labels max to start).
2. Create Custom Field(s):
3. Issue Category (dropdown)
4. Optional: Issue Details (text)
5. Enable or create an AI Task to classify issues.
6. Build dashboards:
7. distribution by issue category
8. trend over time
9. drilldown to conversations

Prompt recommendations

- Provide the allowed label list
- Require the model to select only from the list
- Add Unknown
- Provide a short explanation referencing transcript evidence

Escalation Reason

Escalation Reason identifies why a conversation escalated (or is likely to escalate), such as:

- billing dispute
- technical outage
- policy exception requested
- supervisor requested
- complaint / dissatisfaction

Recommended representation

- Dropdown (controlled list) + optional text details/explanation

Configuration overview

1. Define escalation taxonomy (start small; expand as needed).
2. Create Custom Field(s):
3. Escalation Reason (dropdown)
4. Optional: Escalation Details (text)
5. Enable or create an AI Task.
6. Create dashboards and saved searches:
7. Escalation Reason distribution
8. Saved view: “Escalation Reason != Unknown” or “Escalation Reason = Supervisor requested”

Prompt recommendations

- Use strict allowed labels
- Require evidence-based reasoning
- Return Unknown when escalation is not present or unclear

Define "escalation" based on your business needs. Common approaches:

- **Narrow definition:** supervisor involvement only
- **Broad definition:** includes threats to cancel, public complaints, legal mentions, or explicit dissatisfaction

You can start with a broad definition and use specific reason labels (Supervisor Requested, Cancellation Threat, Complaint, etc.) to segment for reporting.

8.2 Conversation Insights

Call/Conversation Summarization

Summarization produces a concise summary of the conversation to help supervisors and agents understand what happened without reading the full transcript/thread.

Recommended outputs

Common summary fields: - Summary (text, 2–6 sentences) Optional structured companions: - Next Actions (text or dropdown) - Outcome (dropdown) - Key Entities (text)

Configuration overview

1. Create/verify summary Custom Field(s) (text).
2. Enable the Summarization AI Task (prebuilt) or create a tenant task.
3. Apply filters:
4. exclude very short conversations
5. focus on relevant channels/queues
6. Test for quality and consistency.

Prompt recommendations

- Specify length constraints (e.g., 3–5 sentences)
- Include structure (bullets) if desired
- Ask for key details:
- customer intent/problem
- what the agent did
- outcome/resolution
- follow-up steps

Example structure:

```
Return JSON:
{
  "summary": {
    "value": "3–5 sentence summary",
```

```
    "explanation": "Optional: 1 sentence on why this summary is accurate"
  }
}
```

Call Recording supports **Call Summary** as a built-in AI Task type. Summaries are displayed in the Call Details view alongside other call information.

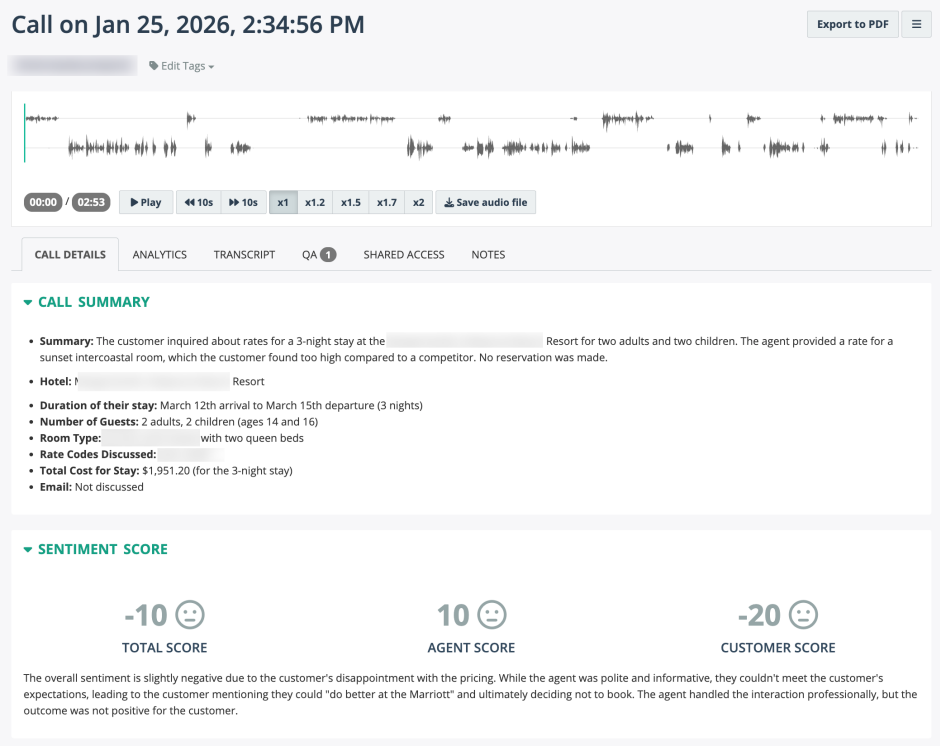

Figure: Call summary displayed in conversation details.

Sentiment Analysis

Sentiment estimates the customer's emotional tone during the conversation (positive/neutral/negative) and can be used for triage and coaching.

Recommended representation options

Choose one:

1. **Dropdown**: Positive / Neutral / Negative / Mixed / Unknown
2. **Numeric score**: -1 to +1 or 1–5 (less common for end users)

Dropdown is recommended for most deployments because it provides clear categories for dashboards and filters (e.g., "Show all Negative sentiment conversations").

Configuration overview

1. Create/verify the sentiment field (dropdown or numeric).
2. Enable the sentiment AI Task (prebuilt) or create a tenant task.
3. Apply filters (optional):
4. skip very short calls
5. Validate using Playground.

Prompt recommendations

- Define label meanings (what counts as "Mixed")
- Require "Unknown" when insufficient evidence
- Provide a short explanation citing transcript evidence

Topic Analysis

Topic Analysis identifies what the conversation is about (topics/themes) to help you understand demand drivers and operational issues.

Recommended representation

- Dropdown (primary topic) or multi-select (multiple topics)

Call Recording supports **multi-label topic classification**—a conversation can be tagged with multiple topics. This provides richer analysis but requires careful taxonomy design to avoid overlap.

Recommendation: Start with a **primary topic** field for stable dashboards. If needed, add multi-topic support later as your taxonomy matures.

Configuration overview

1. Define topic taxonomy (start small; expand)
2. Create Custom Field(s)
3. Enable or create Topic Analysis AI Task
4. Build dashboards and saved searches:
5. topic distribution over time
6. drilldowns to conversations per topic

Prompt recommendations

- Provide allowed topic list
- Require selection only from list
- Provide “Unknown”
- Provide explanation based on evidence

Call Reason and Outcome

Call Reason identifies *why* the customer contacted you. Call Outcome identifies what happened as a result (resolved, follow-up required, escalated, etc.).

These fields are foundational for CX reporting and operational triage.

Recommended representation

- Call Reason: dropdown taxonomy (primary reason)
- Call Outcome: dropdown taxonomy (resolved / unresolved / follow-up / escalated / unknown)
- Optional: explanation fields (or built-in explanations)

Configuration overview

1. Define your reason taxonomy (start with 10–20 reasons)
2. Define your outcome taxonomy (5–8 outcomes)
3. Create Custom Fields (dropdown)
4. Enable the prebuilt AI Task or create a tenant task
5. Validate and roll out dashboards + saved searches

Prompt recommendations

- Provide allowed labels
- Require evidence-based classification
- Include explanation
- Use Unknown when unclear

Call Recording may provide default reason/outcome taxonomies as a starting point. You can customize these by updating the dropdown options in your Custom Fields to match your business terminology. Start with a manageable set (10–20 reasons, 5–8 outcomes) and refine based on actual usage.

8.3 Sales Insights

Lead Score

Lead Score estimates the likelihood that a sales lead will convert (or that the opportunity is strong) based on the conversation.

Recommended representation

Choose one: - Numeric (0–100) with thresholds (Cold/Warm/Hot) - Dropdown (Low/Medium/High)

Recommendation: Use **0–100 numeric** if you want fine-grained ranking and threshold flexibility. Use **Low/Medium/High dropdown** for simplicity and cleaner dashboards.

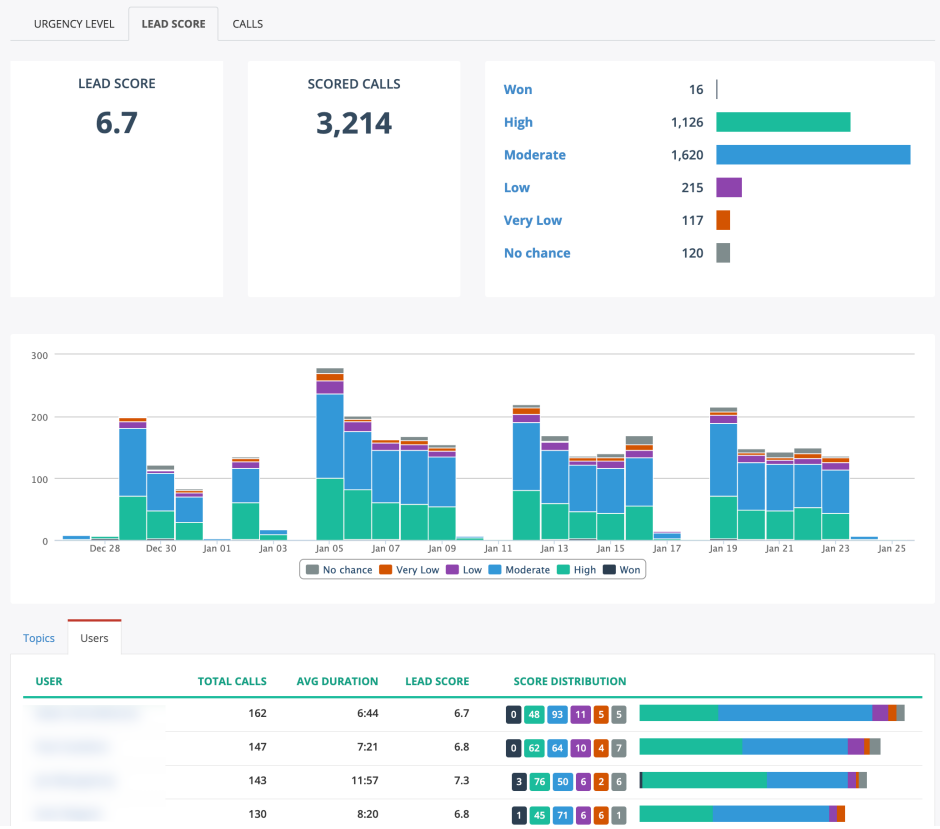

Figure: Lead Score dashboard showing distribution and trend.

Configuration overview

1. Create/verify Lead Score field (numeric or dropdown)
2. Enable prebuilt Lead Score AI Task or create a tenant task

3. Add filters:
4. apply only to sales calls/queues
5. Validate with a sample set and sales stakeholders

Prompt recommendations

- Define what signals a strong lead (budget, timeline, authority, need)
- Require explanation and evidence
- Use Unknown when insufficient information

Top Objections

Top Objections identifies the customer's objections in sales conversations (pricing, features, security, timing, etc.).

Recommended representation

- Dropdown (primary objection) or multi-select (all objections)
- Optional: objection details (text)

Recommendation: Start with **primary objection + details text field**. This keeps dashboards clean while allowing capture of additional context when multiple objections are raised.

Configuration overview

1. Define an objection taxonomy
2. Create Custom Field(s)
3. Enable or create AI Task
4. Build dashboards and saved searches:
5. Objections distribution
6. "Pricing objections this week"
7. "Security objections by segment"

Prompt recommendations

- Provide allowed labels
- Require evidence and brief explanation
- Output Unknown when no objection is present

Competitors Mentioned

Competitors Mentioned detects whether competitors are mentioned in the conversation and (optionally) which competitors.

Recommended representation

Choose one: - Dropdown with competitor names + “None” + “Unknown” - Multi-select competitor list (if supported) - Text field (less ideal for reporting)

Recommendation: Use a **fixed dropdown list** for your known competitors (preferred for dashboards and reporting). Add an "Other" option and optionally a text field for "Other competitor name" to capture competitors not in your list.

Configuration overview

1. Create competitor field(s)
2. Enable competitor task (prebuilt) or create tenant task
3. Filter to sales conversations
4. Validate and build dashboards

Prompt recommendations

- Provide allowed competitor labels
- Require evidence-based extraction
- Return “None” when no competitor mentioned

Deal Amount and Stage

These insights extract sales opportunity details from conversations: - **Deal Dollar Amount** (if discussed) - **Lead/Deal Stage** (e.g., Discovery, Proposal, Negotiation)

Recommended representation

- Deal Amount: numeric (currency) + optional explanation
- Deal Stage: dropdown (controlled stages)

Recommendation: Store deal amount as a **numeric field** in your tenant's base currency. Include an explanation describing the extracted amount and context (e.g., "Customer mentioned a budget of $50,000 for the project").

Configuration overview

1. Create Custom Fields for Amount and Stage
2. Enable or create AI Task
3. Apply sales-only filters
4. Validate for extraction accuracy (amounts can be tricky)

Prompt recommendations

- Extract amount only if explicitly mentioned; otherwise return Unknown/null (as allowed)
- For stage, require selection only from allowed list
- Provide explanations referencing evidence

Next Actions

Next Actions summarizes the agreed follow-up steps from the conversation, such as: schedule a demo - send pricing proposal - follow up next week - customer will provide documents

Recommended representation

- Text field (bullet list) for readability
- Optional structured fields:
- Next Action Due Date (date)
- Next Action Owner (agent/customer)

Configuration overview

1. Create/verify Next Actions field(s)
2. Enable or create AI Task
3. Filter to sales or relevant queues
4. Validate output readability and consistency

Prompt recommendations

- Keep output concise (3–6 bullets)
- Only include actions that are explicitly agreed upon
- Include “None” or empty list when no actions are present (define behavior)

Sales Lost Reason

Sales Lost Reason identifies why a deal was lost (pricing, missing feature, competitor, timing, no budget, etc.). It is useful for win/loss analysis and coaching.

Recommended representation

- Dropdown taxonomy + optional text details

Configuration overview

1. Define a lost-reason taxonomy
2. Create Custom Field(s)
3. Enable or create AI Task
4. Filter to late-stage sales conversations or “lost” tagged interactions (if applicable)
5. Build dashboards and saved searches

Prompt recommendations

- Only set a lost reason when the deal is explicitly lost/ended; otherwise return Unknown
- Provide evidence-based explanation
- Use allowed labels only

Note: Loss reason is typically detected from conversation language (e.g., "we're going with competitor X"). If you have CRM/deal status metadata available, you may use filters to only analyze conversations associated with lost deals for more accurate extraction.

8.4 Auto QA Insights

Auto QA evaluates conversations using your QA scorecard and produces: an overall QA score (and optionally section/question scores) +|- evidence-based explanations for each evaluation

Where Auto QA appears

- Conversation Details (QA results)
- QA dashboards (score distribution and trends)
- Search filters (e.g., QA score < threshold)

Configuration checklist

1. Create or verify your QA Scorecard (sections/questions)
2. Enable/configure the Auto QA AI Task
3. Configure filters (which conversations to evaluate)
4. Test in Playground and calibrate against human QA
5. Roll out dashboards and workflows

See: - **Auto QA (Scorecards and Tasks)** (configuration) - **Playground** and **Validation Checklist** (testing)

Auto QA Dashboard

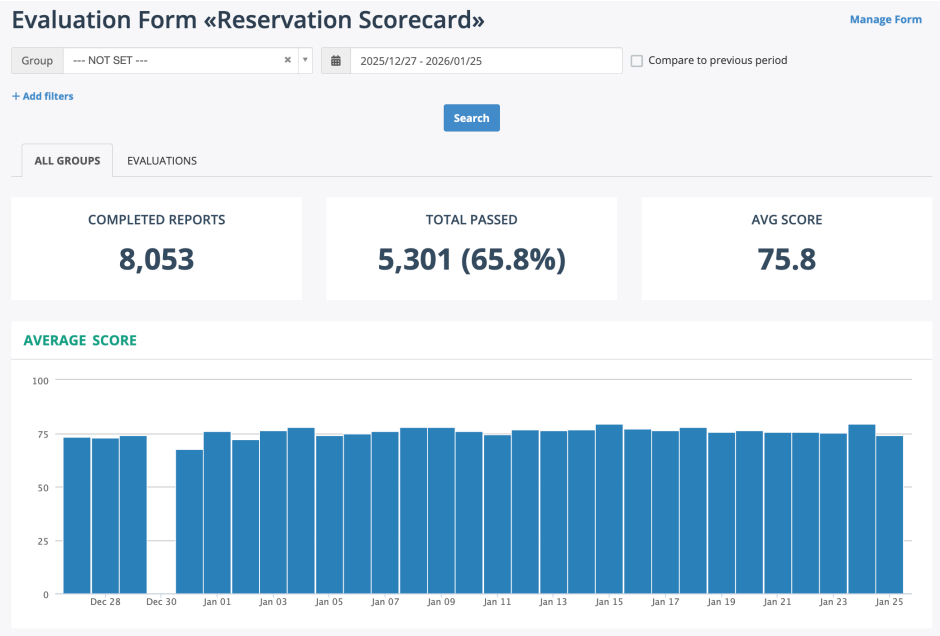

Figure: Auto QA Dashboard showing overall score distribution and trends.

Auto QA Report View

Auto QA results are displayed in a dedicated QA tab on conversation details, showing:

- Overall score
- Section-by-section breakdown
- Per-question results with explanations

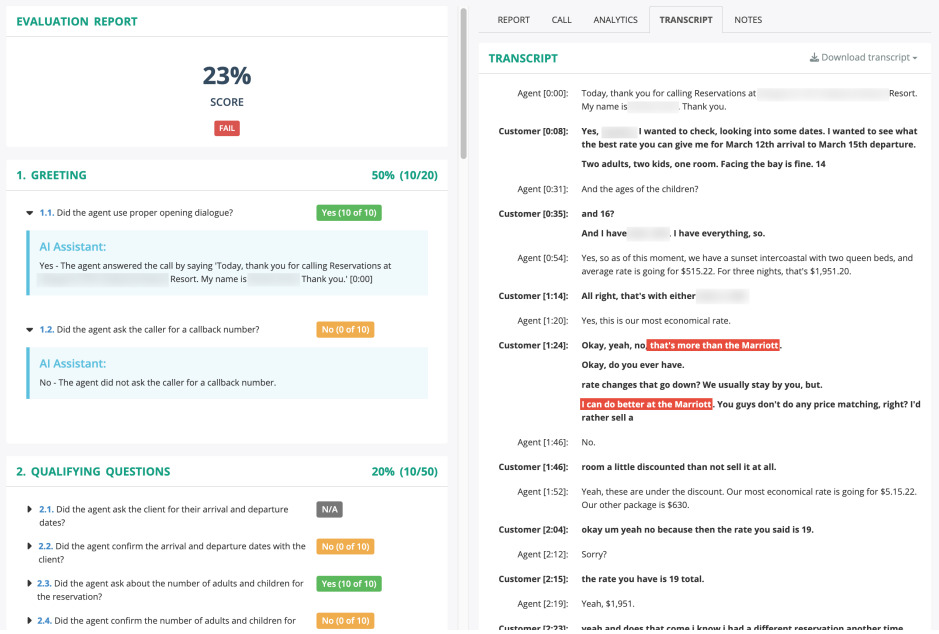

Figure: Auto QA report showing detailed results alongside the conversation transcript.

Users can provide feedback on Auto QA results if they disagree with the scoring:

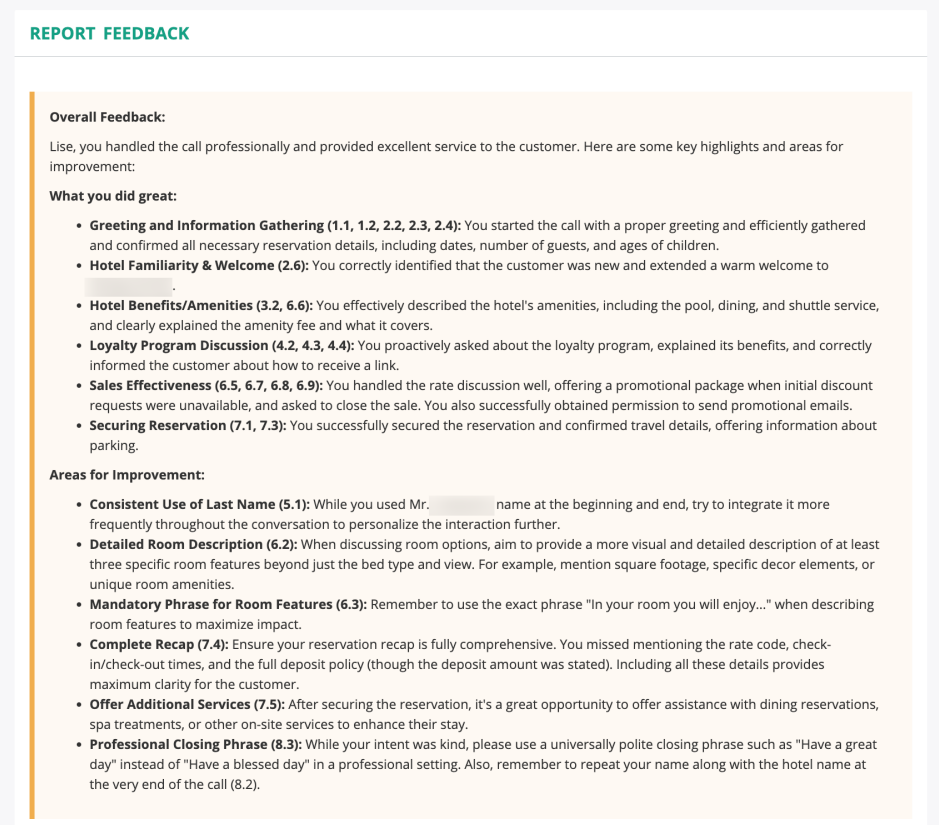

Figure: Feedback option for flagging incorrect results.

8.5 Custom Insight Examples

This page provides example custom insights customers commonly implement. Use these as templates for your own tenant-specific insights.

> For the full configuration workflow, see **Create Custom Insights (Tenant Tasks)**.

Hospitality examples

Reservation Start Date (Date)

- Field type: Date
- Prompt: extract start date if mentioned; otherwise Unknown
- Useful dashboards: arrivals by date, urgent issues for VIP reservations

Reservation Total Nights (Number)

- Field type: integer
- Prompt: extract total nights; require evidence
- Useful searches: stays > 7 nights

VIP Status (Dropdown)

- Values: VIP / Non-VIP / Unknown
- Prompt: classify only if evidence exists; include explanation

Support examples

Issue Category (Dropdown)

- Controlled taxonomy (Billing, Login, Delivery, Account Changes, etc.)
- Avoid free text categories unless you plan manual cleanup

Resolution Status (Dropdown)

- Values: Resolved / Not resolved / Follow-up required / Unknown

Escalation Reason (Dropdown)

- Values aligned to your escalation workflow

Sales examples

Urgency Level (Dropdown)

- Values: High / Medium / Low / Unknown
- Useful for triage and follow-up prioritization

Next Actions (Text)

- Bullet list of agreed follow-ups
- Often paired with a follow-up date

Competitors Mentioned (Dropdown)

- Fixed competitor list for stable reporting

Tip: bundle related insights into a single AI Task

For example, one “Sales Qualification” task can output: - lead_stage - urgency_level - next_actions - objections

This reduces processing overhead and keeps prompts consistent.