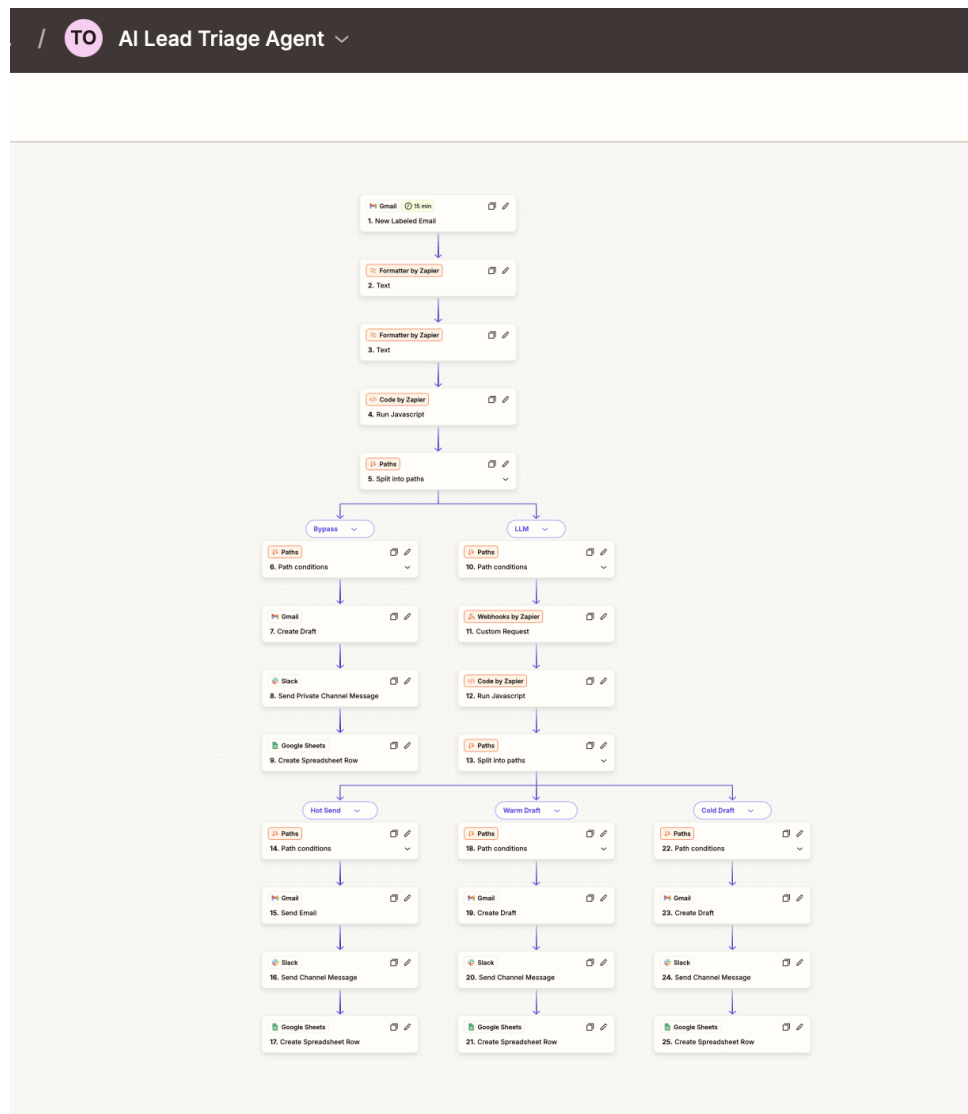


AI Lead Triage + Autoreply Workflow

Zapier workflow for labeled Gmail intake, AI lead classification, Slack alerts, Google Sheets logging, and branch-specific email response handling.

Tools: Zapier, Gmail, Formatter by Zapier, Code by Zapier, JavaScript, OpenAI API, Slack, Google Sheets.



Screenshot 1: Workflow architecture

Multi-step Zapier workflow with controlled Gmail intake, preprocessing, JavaScript normalization, bypass routing, LLM classification, and branch-specific follow-up paths.

Business problem

Inbound sales inquiries often arrive with different levels of urgency, clarity, and fit. Without a triage layer, every message has to be manually opened, interpreted, prioritized, logged, and answered. This makes follow-up inconsistent and can cause high-intent leads to sit in the same queue as unclear or low-priority messages.

What I built

I built a multi-step Zapier workflow that starts from a controlled Gmail intake label, preprocesses the email body, runs a JavaScript normalization and bypass check, then routes the message through either a non-LLM fallback path or an LLM-based lead classification path.

Qualified messages are classified into hot, warm, or cold lead paths. Each path creates the appropriate follow-up action, sends an internal Slack notification, and logs the lead in Google Sheets for tracking.

Workflow logic

Gmail labeled email trigger → text formatting/truncation → JavaScript normalization and bypass logic → path routing → LLM classification → hot/warm/cold branch handling → Gmail response/draft → Slack notification → Google Sheets log.

Implementation details

- Used a dedicated Gmail label as the intake trigger instead of monitoring all inbox activity.
- Added preprocessing steps to clean and limit email body text before classification.
- Used JavaScript to normalize fields such as sender, subject, body, message ID, and date.
- Added a bypass condition so irrelevant or unclear emails could be routed without unnecessary LLM processing.
- Split qualified leads into hot, warm, and cold paths with different follow-up handling.
- Logged each processed lead to Google Sheets for lightweight visibility and auditability.
- Sent Slack notifications so the team could see lead activity without checking Gmail manually.

Design decision

The important design choice was separating deterministic routing from AI classification. The workflow did not treat every inbound email as an AI task. It first checked whether the message belonged in the sales/automation intake flow, then only routed relevant messages into LLM-based classification.

Outcome

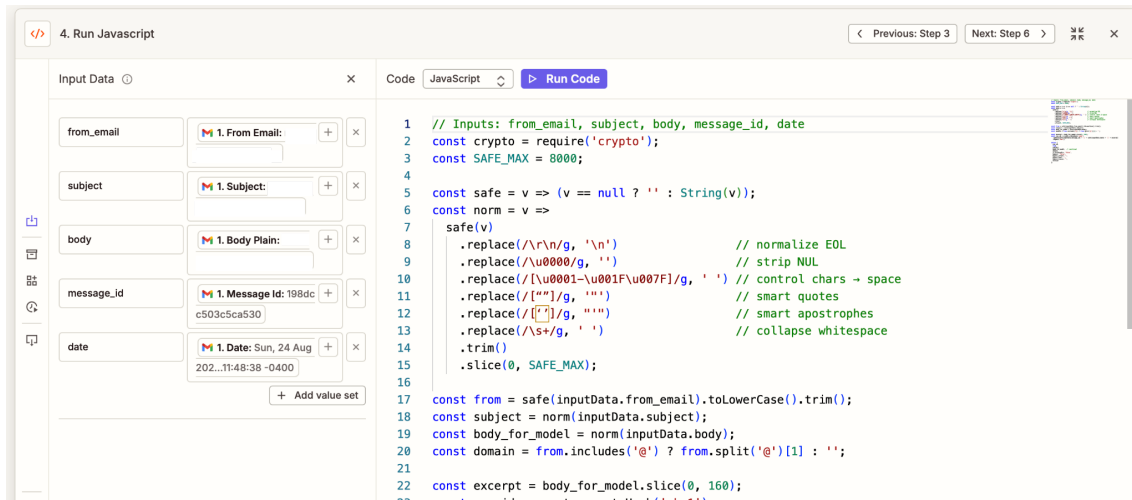
Created a working AI-assisted lead triage prototype that reduced repetitive first-response handling, introduced consistent routing logic, and gave the team a simple operational record of inbound lead activity.

This screenshot shows the configuration interface for the second step, '2. Text'. The interface includes a top navigation bar with 'Setup', 'Configure', and 'Test' tabs, and a 'Next: Step 3' button. The 'Transform' dropdown is set to 'Truncate'. A tooltip explains: 'Limit your text to a specific character length, and delete anything over that.' The 'Values' section contains an 'Input' field. The 'Max Length' is set to 800. The 'Skip Characters' field is empty. The 'Append Ellipsis?' option is set to 'True'. A 'Continue' button is at the bottom.

This screenshot shows the configuration interface for the third step, '3. Text'. The interface includes a top navigation bar with 'Setup', 'Configure', and 'Test' tabs, and a 'Next: Step 4' button. The 'Transform' dropdown is set to 'Replace'. A tooltip explains: 'Replace all instances of any character, word or phrase in the text with another character, word or phrase.' The 'Values' section contains an 'Input' field with a reference to '2. Output:'. The 'Find' field contains the regex pattern '\s+'. The 'Replace' field is empty. A 'Continue' button is at the bottom.

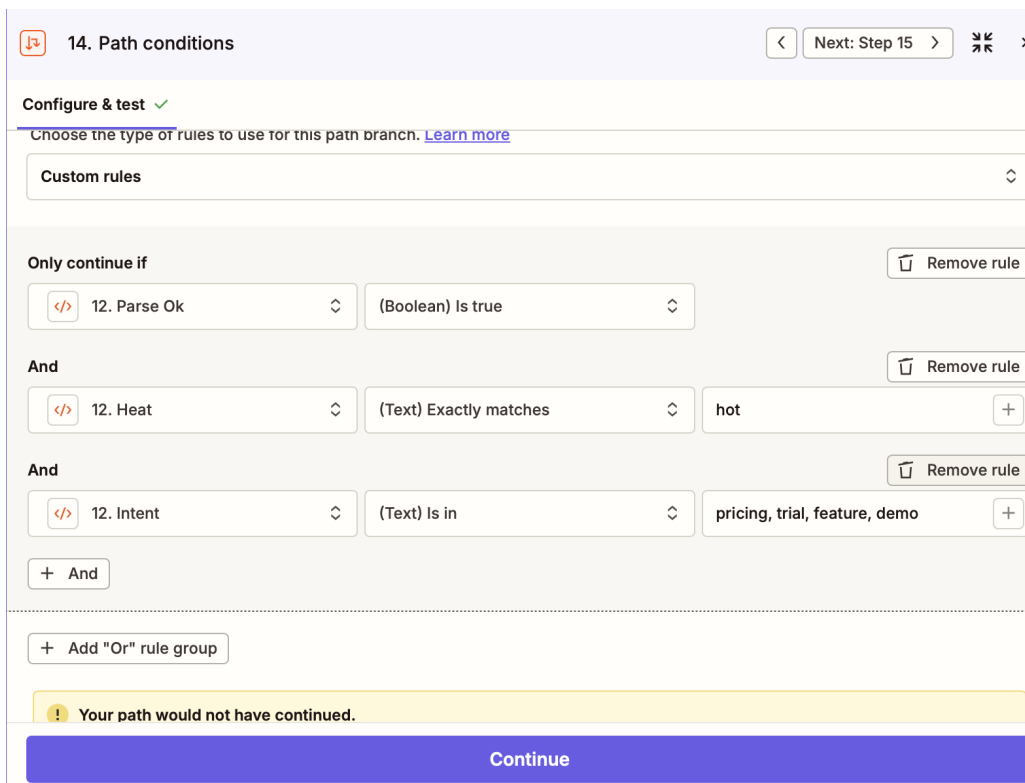
Screenshots 2-3: Input preprocessing

Formatter steps used to truncate and clean inbound email text before downstream routing and classification.



Screenshot 4: JavaScript normalization layer

JavaScript step used to normalize sender, subject, body, message ID, and date; sanitize text; limit payload size; and prepare structured values for routing.



Screenshot 5: Hot lead routing rule

Deterministic path rule for high-intent leads: only continue when parsing succeeds, lead heat is “hot,” and the detected intent matches pricing, trial, feature, or demo.