SYSTEM REPORT · JULY 25, 2026

Mira: The First End-to-End AI Recruiter

Sourcing tools find profiles. Outreach tools send messages. Mira closes the whole loop: from hiring intent to interviews on the calendar, with guardrails at every step.

The Metix AI Team · Metix AI Research

5

specialized sub-agents in one pipeline

1B+

public professional profiles in the talent graph

40%+

gain on our primary business metric

ABSTRACT

Recruiting is less constrained by access to talent than by the operational cost of converting hiring intent into real interviews. Mira, the AI recruiter built by Metix AI, is our answer: a practical, reliable, and efficient agent for proactive recruiting, composed of five specialized sub-agents. Recruiters share their hiring needs and constraints. Mira translates them into an Ideal Candidate Profile, searches a talent graph of over one billion publicly sourced profiles, evaluates fit, engages candidates, and returns only the ones who are interested, together with their latest resumes and available interview slots.

This report describes how Mira is designed from the ground up: the talent graph it runs on, the two models we fine-tuned in house, the responsibilities of each sub-agent, and the evaluation layer that keeps the whole system stable, safe, and honest in production.

KEYWORDS agent systems · recruiting agents · multi-agent pipelines · fine-tuning · talent graph

CONTENTS

01 The bottleneck is not talent

Recruiting today is rarely blocked by access to people. It is blocked by the operational cost of everything between a hiring request and a scheduled interview. Teams translate requirements into search filters, sift through large pools, draft outreach, manage replies, and coordinate calendars. Every step burns hours, invites inconsistency, and slows the cycle down, and it gets worse as volume grows or requirements change mid-search.

Mira attacks exactly that cost. It is an autonomous recruiting system that converts hiring intent into scheduled interviews with minimal recruiter effort. The recruiter provides role context, required qualifications, and constraints: location, seniority, compensation bands, preferences, disqualifiers. Mira returns a curated set of interested candidates, each with a recent resume snapshot and interview time options that fit the recruiter's calendar, plus a transparent rationale for every selection.

> *The deliverable is not a list of profiles. It is a conversation that already wants to happen.*

02 One pipeline, five agents

Mira is implemented as a multi-agent pipeline with explicit interfaces, shared state, and system-level guardrails. Each sub-agent owns a narrow, testable function.

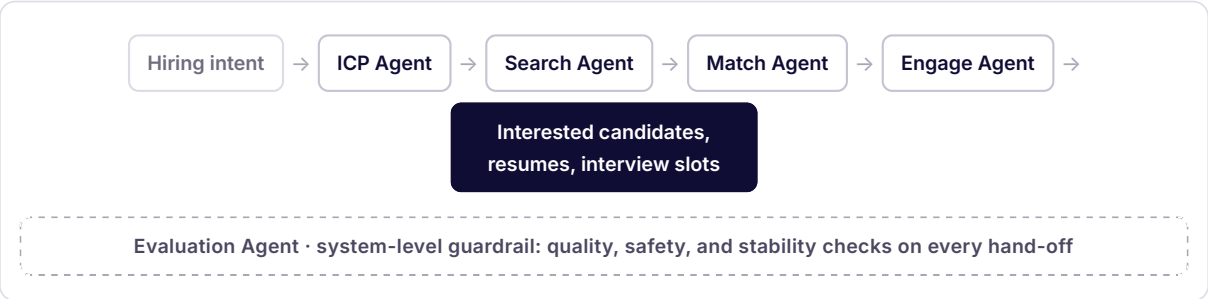

Figure 1 · Mira's agent pipeline. The Evaluation Agent watches every arrow, triggering retries, fallbacks, and human escalation when needed.

ICP AGENT

Translates an ambiguous hiring request into a structured target profile.

HANDS OFF

The contract every downstream agent consumes.

SEARCH AGENT

Runs adaptive retrieval over the talent graph.

HANDS OFF

A candidate slate with full retrieval metadata.

MATCH AGENT

Evaluates and ranks fit against the actual hiring intent.

HANDS OFF

An ordered shortlist where every recommendation carries an auditable reason.

ENGAGE AGENT

Conducts compliant, personalized outreach and multi-turn follow-up.

HANDS OFF

Interested candidates with details and availability.

EVALUATION AGENT

Validates outputs, checks policy compliance, and monitors execution health.

HANDS OFF

Retries, fallbacks, escalations, and release decisions.

The pipeline runs as a closed loop, not a one-way conveyor. Reply rates, interview show rates, recruiter selections, and downstream hiring signals flow back to refine ICP extraction, retrieval strategy, scoring calibration, and message generation. Every search teaches the system something about the next one.

03 One billion profiles, all public

Mira runs on one of the largest talent datasets ever assembled for recruiting: a unified talent graph covering over one billion professional profiles, continuously refreshed, where each record brings together career history, skills, projects, publications, and public professional signals into one coherent picture of a person.

OVER

1B

professional profiles in the unified talent graph, continuously refreshed

EXACTLY

100%

of the graph built from publicly accessible sources on the open Internet

Every record is built exclusively from information that is publicly available on the Internet: the professional presence people choose to publish across professional networks, developer and open-source communities, research repositories, and personal websites. We do not access private APIs, restricted content, or non-public user data.

Compliance is a design constraint, not an afterthought. The data pipeline is built to comply with major privacy regulations, including GDPR and CCPA, so that everything Mira knows about a candidate stays lawful, transparent, and privacy-preserving. Scale matters in recruiting, but only when it comes with trust.

04 Why general-purpose models were not enough

Mira's core runs on two models we fine-tuned in house: an embedding model for semantic retrieval, and an LLM for the hiring workflows around it.

General-purpose embeddings kept failing us in the same ways. They lean on superficial keyword overlap and miss hiring semantics: role-specific terminology, fine-grained differences between adjacent roles, and relevance calibrated by seniority and hard constraints. Three failures we saw constantly:

FAILURE MODE	WHAT THE BASE MODEL DID	THE COST
Abbreviations, polysemy	Confused "SOC 2" compliance experience with unrelated "SoC" hardware work	Irrelevant candidates surface in the slate
Blurred role boundaries	Matched an enterprise Product Marketing Manager role, focused on positioning and sales enablement, to brand marketing resumes on shared generic vocabulary	Precision collapses between adjacent roles
Miscalibrated seniority	Ranked junior "Go" profiles highly for a senior distributed systems role because the keyword appears	Keyword presence outweighs actual fit

Table 1 · Three retrieval failures that motivated fine-tuning our own recruiting-native embedding model.

So we fine-tuned our own recruiting-native embedding model on large-scale, hiring-specific training data, so that similarity reflects recruiting intent rather than topical resemblance. Higher retrieval precision means far fewer profiles to re-rank, review, and message per qualified interview.

OVER
40%
improvement on our primary business metric versus a general-purpose embedding baseline

MORE THAN
90%
savings in operational cost per qualified, interested candidate in production-like evaluation

The LLM side has the same domain gap. Job titles that look identical imply very different scopes across industries and regions. Candidate profiles are full of long-tail skills, abbreviations, and implicit signals that need consistent normalization. And a production system needs predictable behavior under strict business constraints: stable ranking across refreshes, controllable recall-precision tradeoffs, calibrated confidence for screening decisions. Fine-tuning lets us encode our taxonomy, evaluation rubric, and feedback signals directly into the model. It improves retrieval relevance and fit-judgment consistency, cuts our reliance on brittle prompt tricks, and lets smaller, faster models hit production quality at a fraction of the cost.

05 Intake that earns every question

The ICP Agent's first job is to capture what the recruiter actually needs, not the surface-level job title. Mira runs a short, structured intake: business context, must-have constraints, nice-to-have preferences, compensation range, location and work model, and the outcomes the hire is expected

to deliver in the first 90 days. From this it builds the Ideal Candidate Profile: target role variants, seniority expectations, domain keywords, disqualifiers. The ICP becomes the shared contract for the rest of the pipeline, so retrieval, ranking, screening, and outreach all optimize around the same intent instead of drifting apart agent by agent.

The tension is that every extra intake question raises drop-off. Mira is built around that tradeoff: instead of a long form up front, it keeps the initial setup deliberately short, gets to real candidates quickly, and asks a follow-up only when the answer would genuinely change the search. The conversation feels fast because every additional question pays for itself with a visible improvement in results.

06 Search as a closed loop

Hiring intent is almost always underspecified at the start, and a single-pass query returns either too many weak matches or too few candidates. So the Search Agent is built as an adaptive retrieval engine, not a static query generator. Every round of retrieval produces evidence about what the recruiter actually meant and what the market can actually supply.

At the core, it translates the recruiter's intent into an executable retrieval plan with an explicit strategy for recall versus precision, deciding signal by signal how strictly each requirement should be interpreted. It also carries the hiring-specific edge cases that break naive search: title inflation, cross-industry title mismatch, multi-role careers, and the long tail of skill synonyms and abbreviations.

After each execution, the result distribution is feedback, not a final answer. When the pool comes back too small, the agent broadens in a controlled way; when it comes back too large or low quality, it tightens. Each revision is targeted, minimal, and grounded in what the previous round showed, so the loop converges quickly instead of oscillating, and it never drifts away from the role the recruiter actually described.

Stability is enforced throughout. Hard constraints and soft preferences never blur into each other. Every revision is logged, so the system can explain why a query changed and reproduce any result. Sensitive attributes sit behind conservative guardrails. And when a request is internally contradictory or the data is simply sparse, the agent does not keep broadening forever. It surfaces the conflict, proposes the smallest revision that restores feasibility, and escalates to a human when needed.

07 A grader, not a score

"Fit" is rarely a fixed checklist. The same requirement behaves like a hard constraint in one role and a soft preference in another, and similar titles imply very different scopes across companies and industries. So the Match Agent is built as a dynamic grader rather than a black-box scorer: its job is a structured hiring judgment backed by evidence, not a floating number.

It consumes the ICP as the evaluation contract and the Search Agent's retrieval evidence as grounding, then decides how to grade. Hard requirements become pass-fail gates wherever possible, so a strong overall profile cannot slide past a missing critical constraint. Preferences become weighted features that shape ordering and tradeoffs.

The weights are not static. The grader adapts to the role's context and to what the market can actually supply: it reads the candidate distribution and adjusts how strictly each dimension is weighed, so a crowded market gets filtered harder and a rare role is judged with more nuance. Controlled reweighting that preserves the meaning of the role, never blind broadening.

The output is built to be acted on. An overall grade plus dimension-level grades (role-relevant experience, core skill coverage, domain alignment, seniority and impact, collaboration signals, risk factors), each with citations to the supporting evidence and explicit uncertainty where the profile is incomplete. Decisions triage into strong fit, potential fit, or not fit, with concise reasons for both acceptance and rejection.

The grader also distinguishes claims from proof, "worked on" versus "owned", and demands specific evidence for leadership, scope, and domain expertise. For borderline cases it prefers a lower confidence plus a request for targeted missing information over invented detail. Over time, recruiter selections and downstream outcomes recalibrate which evidence actually predicts interview success for each role family.

08 Outreach as an operational system

The hardest part of engagement is not writing one good message. It is staying consistent, relevant, and compliant across many candidates at once, while adapting to each person's background and the realities of deliverability.

For the first touch, the Engage Agent combines three inputs: the ICP (what matters most), the Match Agent's output (why this specific person fits), and the candidate's own context (recent role, projects, skills, location). From those it selects an angle: direct alignment with a recent project, a domain problem the role exists to solve, or a career step that matches the candidate's trajectory. No generic templates. Every message carries concrete, evidence-grounded personalization and stays concise enough to survive a real inbox.

Then engagement becomes a stateful, multi-turn process. If the candidate is interested, the agent collects the minimum needed to move fast: an updated resume, current location, work authorization, compensation expectations, interview availability. If they raise concerns, it clarifies at the right level of detail without overpromising. If they are lukewarm, it tests an alternative value proposition, remote flexibility, growth scope, mission fit, while respecting boundaries. If they go silent, it schedules spaced, varied follow-ups and stops after a defined limit, because brand reputation and deliverability outlast any single role.

Compliance is enforced, not hoped for: no sensitive-attribute inference, no discriminatory language, alignment with jurisdictional requirements, unsubscribe handling, contact frequency

limits, and suppression rules. The agent also monitors bounce, complaint, and low-reply patterns, then adapts sending strategy and content to protect domain health. Outreach here is an operational system, not copywriting.

09 The agent that grades the agents

In a multi-agent workflow, a small change in one component can quietly regress another: a retrieval tweak that lifts recall but degrades outreach relevance, a prompt edit that makes grading more verbose but less consistent. The Evaluation Agent exists to measure those effects early, quantify the tradeoffs, and block releases that would hurt real hiring outcomes.

It evaluates Mira at two levels. At the component level, each agent gets task-specific datasets and rubrics: intent-to-retrieval correctness for Search, calibration and consistency for Match, personalization quality and compliance for Engage. At the end-to-end level, it simulates realistic hiring flows from intake to shortlist to engagement, scored on business-aligned metrics: qualified candidate yield, time to shortlist, reply rate, and interview conversion, with cost and latency tracked alongside.

It also owns the evaluation corpora: gold-labeled examples, hard negative pairs, long-tail role variants, and deliberately adversarial cases (underspecified requirements, conflicting constraints, noisy profiles, misleading keyword overlaps), because these are exactly where multi-agent systems fail in non-obvious ways. The output of a run is never a bare pass or fail. It is a structured report on what changed, where the failure modes appear, and which signals drove the regression.

Once changes reach production, evaluation continues online: guardrail metrics for distribution shift in retrieved candidates, instability in grading, sudden drops in reply quality, and cost spikes, with holdouts and traffic splits for statistically grounded comparison. On significant degradation or safety risk, it can recommend rollback, tighten thresholds, or route more cases to conservative fallbacks. And it closes the learning loop by normalizing recruiter selections, rejection reasons, candidate replies, and interview results into labels that improve both future evaluation sets and the models themselves.

How we think about this layer in depth is its own note: Agent Evaluation, Done Right.

10 Closing thoughts

Mira turns hiring requirements into real interviews by automating the operational work that slows teams down: candidate search and shortlisting, outreach copywriting and messaging, follow-up coordination and scheduling. Building it convinced us of two things. First, recruiting is a domain where general-purpose models are not sufficient: fine-tuning for hiring semantics changed the economics of the entire pipeline. Second, autonomy is only useful when it is paired with explicit guardrails and continuous evaluation, so that speed never comes at the price of trust.

Autonomy is easy to demo. Making it something a recruiter can rely on every single day is the actual work, and that is the part Mira was built around.

Further reading

Agent Evaluation, Done Right JULY 2026

Why the durability of a production agent depends on its evaluation system, and how we built ours for Mira: research-style testing, layered scoring, and evaluation wired into CI.

Mira-Embeddings-V1: Domain-Adapted Semantic Reranking for Recruitment APRIL 2026

The retrieval model behind the numbers in section 04: LLM-synthesized supervision and boundary-aware reranking for candidate retrieval.

Metix AI

Metix AI is an AI-native hiring platform. Interview-ready candidates in days, not weeks.
8 The Green, Ste A, Dover, DE 19901, USA · metix.ai · contact@metix.ai