Autonomous Oncology Market Intelligence: A Multi-Vertical Knowledge Graph with Agent-Directed Gap-Filling and Prediction Modelling

Raminderpal Singh
raminderpal@hitchhikersai.org
raminderpalsingh.com | scienceclaw.ai
v1.0 (March 2026)

Abstract

The US oncology market spans eight analytically distinct verticals — academic research, drug discovery, clinical trials, manufacturing, sales, distribution, FDA regulatory, and government policy — each maintained in separate data silos with no integrated cross-vertical view. We describe Thread 5, an autonomous market intelligence system that constructs a unified knowledge graph (KG) across all eight verticals from free public APIs, uses OpenClaw agents to identify structurally interesting emerging edges, direct web-based gap-filling of missing features, train prediction models, and generate explainability narratives. The KG uses a three-tier evidence structure (Tier 1: structured API data; Tier 2: agent-extracted and web-sourced data; Tier 3: deterministically inferred edges) with confidence scores computed by deterministic rules, never by language model self-assessment. Four sklearn prediction models output approval probability, market entry timing, revenue trajectory index, and disruption risk per drug-indication pair, with permutation importance and coefficient contributions as explainability mechanisms. A weekly eight-stage OpenClaw agent pipeline autonomously discovers novel KG patterns, fills data gaps through targeted web search, evaluates model quality, and publishes results to scienceclaw.ai. The target user is a senior market research analyst in a large pharmaceutical company seeking integrated cross-vertical signal detection that would otherwise require manual synthesis across heterogeneous sources.

Keywords: oncology market intelligence, knowledge graph, multi-vertical integration, autonomous gap-filling, prediction modelling, drug approval probability, disruption risk, OpenClaw, explainability, US oncology market

1. Introduction

Senior market research analysts in large pharmaceutical companies operate across a landscape of fragmented data sources: NIH grant databases, ChEMBL compound registries, ClinicalTrials.gov, FDA approval records, CMS reimbursement data, and Congressional legislation. No integrated system continuously synthesises these signals or detects when cross-vertical patterns — for example, a simultaneous surge in NIH funding, first-in-class Phase 3 trial registration, and FDA Breakthrough designation for the same indication — suggest an emerging inflection point.

Commercial market intelligence platforms (IQVIA, Evaluate, GlobalData) provide comprehensive coverage but require expensive subscriptions and do not surface cross-vertical structural anomalies in real time. This paper describes Thread 5, an autonomous system built on the ScienceClaw platform that addresses the

cross-vertical integration problem using only free public APIs.

Thread 5 shares the ScienceClaw server and OpenClaw orchestration framework with Threads 2 (Trial Intelligence anomaly detection), 3 (Oncology Insights KB-building), and 4 (Trial Knowledge Graph). It runs on a weekly cycle staggered before Thread 2 to avoid concurrent LLM calls, and cross-references Thread 2 findings as Tier 1 evidence for trial signals.

2. Design Principles

Eight-vertical integration, not landscape summarisation. The system's value is in detecting cross-vertical structural patterns invisible in any single data source. Intra-vertical summaries are a by-product, not the primary output.

Deterministic computation, agentic reasoning. All data collection, KG writing, feature engineering, and scoring are handled by deterministic Python scripts. OpenClaw agents are used exclusively for reasoning tasks: identifying interesting edges, prioritising data gaps, evaluating web search results, and assessing model quality.

Confidence scores are never LLM-assigned. All confidence scores in the KG are computed by deterministic rules from schema constants. The agent nominates predictor types and mechanisms; it never assigns numeric confidence.

Single writer principle. market_kg_writer.py holds exclusive write access to market_kg.json. All other scripts submit to a staging directory or proposals queue.

Honest coverage disclosure. Manufacturing, distribution, and sales verticals have structural gaps from public data. The system labels coverage quality explicitly in every output and does not attempt to compensate with fabricated values.

Graceful degradation. A failure in any single vertical produces a partial KG, not a pipeline abort. The KG writer loads whichever staging files exist.

3. Data Sources and Integration

Thread 5 integrates eight verticals, each accessed via free public REST or GraphQL APIs. No commercial data subscriptions are used. Coverage quality varies substantially across verticals and is disclosed explicitly to the user.

Table 1. Data sources by vertical.

Vertical	Primary APIs	Node types produced	Coverage
1. Academic research	PubMed E-utilities, NIH Reporter v2	Publication, Grant, Institution	High
2. Drug discovery	ChEMBL REST, Open Targets GraphQL	Drug, Target	High
3. Clinical trials	Thread 2 findings.json, Thread 4 KG (Sponsor only)	KC2 (Sponsor)	Inherited
4. Manufacturing	openFDA drug enforcement, Drugs@FDA application_events	Manufacturer, Recall_event	Low — existence only
5. Sales	CMS Part D spending dashboard (annual snapshot)	Spending_snapshot	Medium — 2yr lag
6. Distribution	HRSA 340B OPA covered entities, NPI registry	Distributor_entity	Low — structural only
7. FDA regulatory	openFDA Drugs@FDA (NDA/BLA approvals), Orange Book	Approval, Exclusivity, Patent	High

8. Government policy	Federal Register API, Congress.gov API	Policy_act, Regulation, Agency	Medium — text-heavy

3.1 ClinicalTrials.gov and Thread Integration

Thread 5 does not independently query ClinicalTrials.gov. Trial data is inherited from Thread 2 (findings.json) and Thread 4 (kg.json) via a zero-API-call cross-reference script. This prevents duplicate API load and ensures Thread 5 trial nodes are consistent with Thread 2's anomaly classifications. ANOMALY_SIGNAL edges from Thread 2 USEFUL-labelled findings become Tier 1 evidence in the market KG.

3.2 Federal Register and Congress.gov

Policy collection is the most complex vertical. The Federal Register API is queried for FDA guidance documents and CMS rules mentioning oncology-relevant terms. Congress.gov is queried for drug pricing and oncology-related legislation. Both sources produce Tier 1 metadata nodes (title, date, agency, bill stage) and flag documents for Tier 2 GLM-5 extraction of indication-specific links. A Congress.gov API key is required (free, issued instantly). The containAll field in the Federal Register API does not support OR syntax — individual terms must be looped and deduplicated.

3.3 CMS Part D Data Lag

CMS Part D drug spending data has a ~2-year publication lag and covers Medicare-reimbursed expenditure only — not total US oncology market revenue. All spending nodes are labelled as directional indices, not revenue forecasts, and the lag is displayed prominently in the predictions output.

4. Knowledge Graph Architecture

The market KG is a node-edge graph serialised to market_kg.json. It uses the same three-tier evidence structure as Thread 4, adapted for market verticals.

Table 2. Evidence tier structure.

Tier	Confidence range	Source	Examples
Tier 1	0.85–1.0	Structured API responses	ChEMBL, openFDA, PubMed, NIH Reporter, Thread 2 findings
Tier 2	0.50–0.75	Agent-extracted from text + web search	Federal Register policy links, gap-fill web search proposals
Tier 3	0.35–0.65	Deterministically inferred edges	COMPETES_WITH, BIOSIMILAR_OF, SUPPLY_RISK_FOR, DOMINANT_IN

4.1 Node and Edge Types

The KG contains 14 node types spanning all eight verticals: Indication, Drug, Approval, Sponsor, Grant, Publication, Target, Institution, Policy_act, Regulation, Agency, Recall_event, Spending_snapshot, and Distribution_entity. Indication nodes are the structural anchors — 16 oncology indications are pre-registered as Tier 1 nodes with EFO identifiers, providing a stable foundation for cross-vertical edge aggregation.

Canonical edge types include APPROVED_FOR, IN_PHASE (Drug to Indication), TARGETS_INDICATION (Grant to Indication), HAS_APPROVAL (Drug to Approval), FILED (Sponsor to Approval), STUDIES_INDICATION (Publication to Indication), ANOMALY_SIGNAL (Trial to Indication, from Thread 2), and HAS_CMS_SPENDING (Drug to Spending_snapshot).

4.2 Tier 3 Inference Rules

Six deterministic inference rules generate Tier 3 edges from patterns in the Tier 1/2 graph. Confidence scores are computed from schema constants, not LLM assessment.

Table 3. Tier 3 inference rules.

Rule	Edge type	Condition	Confidence
T3-01	COMPETES_WITH	Two sponsors share ≥1 approved indication	0.55
T3-02	BIOSIMILAR_OF	Same generic name, different application numbers	0.62
T3-03	PRECEDES_TRIAL	Grant indication matches trial indication, grant fiscal_year ≤ 2022	0.58
T3-04	SUPPLY_RISK_FOR	Class I/II recall affects drug linked to indication	0.60
T3-05	POLICY_RISK_FOR	Policy_act title contains drug name fragment (≥5 chars)	0.52
T3-06	DOMINANT_IN	Sponsor has ≥3 approved drugs in one indication	0.65

4.3 KG Diff and Temporal Tracking

Each week, market_kg_diff.py computes the structural difference between the current KG and the prior week's backup. The diff captures: new nodes by type, new edges by type, rising-degree nodes (nodes whose connection count increased most), new cross-vertical edges (edges connecting nodes from different API sources), and new Tier 3 inferred edges. This diff is the primary input to the Edge Discovery agent, which reasons about which structural changes represent meaningful market signals.

5. OpenClaw Agent Pipeline

The weekly pipeline consists of eight jobs registered in OpenClaw's cron scheduler. Four jobs are deterministic script runners (agentId: trials-oncology). Four are genuine reasoning agent turns (agentId: main) that cannot be replaced by deterministic scripts.

Table 4. Weekly pipeline — Thread 5 (Sunday 06:00–11:30 UK).

Time (UK)	Job	Type	What happens
06:00	thread5-01-collect	Script runner	8 vertical collection scripts run sequentially; KG writer merges staging files; KG c
08:30	thread5-02-edge-discovery	Agent (main)	Agent reads KG diff + centrality scores; reasons about which new cross-vertical
09:00	thread5-03-gap-identify	Agent (main)	Agent runs gap report script; reasons about which missing features most impair
09:30	thread5-04-gap-fill	Agent (main)	Agent searches web for prioritised gaps; evaluates source quality; submits propo
10:15	thread5-05-models	Script runner	Feature engineering (9 features) and sklearn model training; predictions.json wri
10:30	thread5-06-lead-lag	Script runner	Cross-correlation of annual time series; r > 0.5, n ≥ 8 threshold; lead_lag_results
11:00	thread5-07-model-review	Agent (main)	Agent evaluates model quality; writes explainability narrative; runs HTML genera
11:30	thread5-08-weekly-digest	Agent (main)	Agent composes weekly digest email to operator summarising KG changes, top

5.1 Edge Discovery Agent

The Edge Discovery agent is the most novel component of Thread 5. It reads the weekly KG diff and centrality scores and reasons about which structural changes represent meaningful market signals. Four discovery criteria are defined: (A) rising connectivity — a drug node whose degree increased by ≥3 this week across multiple verticals; (B) new cross-vertical bridge — a node that now connects two previously unlinked verticals; (C) high-confidence Tier 3 inference (confidence ≥0.60 for DOMINANT_IN or COMPETES_WITH); and (D) anomaly signal convergence — a drug appearing in both Thread 2 trial findings and a new FDA

Approval in the same indication.

The agent writes edges_of_interest.json: a ranked list of drug×indication pairs with the discovery criterion met, prediction target, primary data gap, and a one-sentence causal mechanism. This output directly drives gap prioritisation in the next stage.

5.2 Gap Identification and Fill Agents

The Gap Identify agent cross-references edges_of_interest.json with the deterministic gap report (null feature counts per drug×indication pair) and reasons about which missing features most impair prediction quality for the highest-interest pairs. It classifies each gap as web_search (findable on current public web), api_fetch (specific free API available), agent_inference (derivable from adjacent KG nodes), or unfillable (genuinely unavailable). A maximum of 25 prioritised gaps are passed to the Gap Fill agent.

The Gap Fill agent executes web searches, reads source pages, extracts numeric feature values, and submits structured proposals with source URLs, snippets, and relevance scores (agent-assessed, 0.0–1.0). Proposals are validated by market_tier2_commit.py: confidence is computed deterministically as source_quality × relevance_score, and proposals below 0.45 are rejected. Validated proposals are written as Tier 2 Feature_value nodes and the KG writer re-runs to merge them.

5.3 Model Review Agent

After the prediction models run, the Model Review agent evaluates output quality against three criteria: CV AUC adequacy (threshold 0.65 for the logistic model), class balance (both classes ≥10%), and feature dominance (whether a single feature drives all predictions). It checks whether the models are capturing the signals identified by the Edge Discovery agent and flags misalignments (typically caused by missing features). It then writes a plain-English narrative for the predictions page and up to three feature improvement proposals for the next cycle.

6. Predictor Identification

Predictor identification runs in three stages: deterministic graph centrality scoring, temporal lead-lag cross-correlation, and GLM-5 classification with Kimi K2.5 verification.

6.1 Graph Centrality Scoring

PageRank and betweenness centrality are computed on the full market KG using a pure stdlib implementation (no networkx import required in this stage, though networkx 3.6.1 is available in the venv). A temporal degree delta — the change in total degree since the prior week — is also computed. These three metrics are combined into a composite score (PageRank 40%, betweenness 40%, delta 20%) to produce a ranked list of top-40 predictor candidates. Indication nodes are excluded — they are prediction targets, not predictors.

6.2 Lead-Lag Cross-Correlation

Four lead-lag relationships are tested on annual time series extracted from the KG: NIH grant funding → FDA approval count (1–5 year lags), publication velocity → approval count (2–6 year lags), grant funding → publication count (1–3 year lags), and approval count → CMS spending (1–3 year lags). Only correlations passing $r > 0.5$ with $n \geq 8$ data points are reported. Confidence is computed deterministically: $r \times \min(n, 24) / 24$. In early deployment weeks this threshold eliminates most results — this is correct behaviour, not a failure. The predictor set grows as weekly data accumulates.

6.3 GLM-5 Classification with Kimi K2.5 Verification

Top centrality candidates and significant lead-lag results are passed to GLM-5 for classification as leading, lagging, or coincident indicators, with target outcome and mechanism. GLM-5 output is validated against a strict schema before Kimi K2.5 verifies each classification. Records rejected by Kimi are excluded from the predictor registry. Numeric confidence scores in the predictor registry are always derived from Stage 2 correlation results, never from GLM-5 self-assessment.

7. Prediction Models

Four prediction models are trained weekly using scikit-learn. All models use tabular features derived deterministically from the KG. The M1 16GB hardware constraint (with four concurrent threads) requires that all models be lightweight; no neural networks or PyTorch are used.

Table 5. Feature matrix — 9 features per drug×indication pair.

Feature	Derived from	Reliability
NIH funding momentum (USD/yr)	NIH Reporter grant award trend slope, 24-month window	High
Publication velocity (CAGR)	PubMed publication count, 3-year compound annual growth rate	High
Pipeline density (trial count)	ClinicalTrials.gov Phase 2/3 active trial count via Thread 2	High
Trial anomaly rate	Thread 2 USEFUL-labelled findings count per indication, 90-day proxy	High
Regulatory momentum (approval count)	openFDA NDA/BLA total approvals for this drug	High
Patent cliff proximity (years)	FDA Orange Book patent expiry year (deferred — not yet accessible)	Not available
Policy risk score (normalised)	Congress.gov drug pricing bill count weighted by stage	Medium — global, not indication-specific
CMS spend trajectory (USD/yr)	CMS Part D spending trend slope	Low — 2yr lag; directional only
Shortage risk score (count)	openFDA Class I/II recall count for this drug	Medium

Table 6. Prediction models.

Model	Algorithm	Target variable	Label construction	Expected performance
Approval probability	Logistic regression (L2, C=1.0)	P(recent NDA/BLA approval, post-2018)	In-sample from KG approval codes; recent vs older drug split	CV AUC ~0.68–0.75 with current features
Market entry timing	Empirical hazard (no sklearn)	Distribution of FDA approval years by indication	Historical approval year distribution from KG	Wide CI; treat as distributional, not point estimate
Revenue trajectory index	Ridge regression (alpha=1.0)	Projected CMS spending growth (direction)	CMS Part D 2015–2022 (limited, 2-yr lag)	Low — directional signal only
Disruption risk index	Random forest (50 trees, depth 4)	Composite: shortage + competitive entry signals	Observable KG signals; no external ground truth	Relative index, not a validated probability

7.1 Label Construction

All training labels are constructed in-sample from the KG. The approval model uses a recent/older split (post-2018 = 1, older = 0) rather than a has_approval label — the latter produces all-positive labels since every drug in the KG came from Drugs@FDA. The post-2018 threshold creates a meaningful split: next-generation targeted therapies and immunotherapies versus older chemotherapy agents. This is an in-sample split; CV AUC is indicative only. True generalisation requires held-out data not available from free public sources.

7.2 Explainability

Two explainability methods are used. For logistic and ridge regression: signed coefficient × feature value per prediction row (direct per-feature contribution). For the random forest: permutation importance with n_repeats=50 (sklearn's permutation_importance — safe on M1 16GB for small feature matrices). SHAP is not used: it requires a separate install and adds memory overhead unjustified for 9 features. Per-prediction contribution bars are displayed in the market-predictions.html drill-down panel.

8. Output and Deployment

Thread 5 produces three static HTML pages deployed to scienceclaw.ai via Vercel CLI. All pages are generated by Python scripts that embed JSON data at generation time — no runtime API calls in the browser. A standalone landing page (scienceclaw-market.html) links to both output pages and is accessible from the main site navigation.

Table 7. Output pages.

Page	URL	Content
Landing page	scienceclaw.ai/scienceclaw-market.html	KG and prediction stats, vertical coverage summary, pipeline diagram, links to c
Market KG	scienceclaw.ai/market-kg.html	Force-directed network graph (vanilla JS canvas, no D3); node type filter pills; s
Market predictions	scienceclaw.ai/market-predictions.html	Sortable table of 152+ drug×indication predictions; per-row drill-down with featu

8.1 Predictions Page Design

The predictions page is designed for a sceptical, experienced user. Every prediction displays its 80% confidence interval (bootstrap n=200), its primary driver features, and a data freshness badge per vertical. The caveat panel listing model limitations is always visible and not collapsible. The agent-authored narrative (from the Model Review agent) appears above the table and is regenerated weekly. Hiding uncertainty or burying caveats would undermine trust with the target user type.

8.2 Knowledge Graph Visualisation

The KG visualisation uses a force-directed simulation with Verlet integration (280 tick settlement, then all nodes pinned). Nodes are sized by confidence and coloured by type. COMPETES_WITH edges are excluded from rendering — at 2,400+ they make the graph unreadable. Tier 3 edges render as dashed 0.5px lines at reduced opacity. After settlement, dragging a node pins it in place; the simulation does not restart. Canvas pan and scroll-to-zoom are supported. The visualisation pattern matches Thread 4's KG tab.

9. System Architecture

Table 8. Technology stack.

Component	Demo instance	Role
Agent framework	OpenClaw	Multi-agent orchestration, cron, session isolation
Analysis LLM	GLM-5:cloud (Ollama)	Edge discovery, gap identification, gap fill, model review reasoning
Verification LLM	Kimi K2.5 (Ollama Cloud)	Predictor classification verification, weekly digest verification
Prediction models	scikit-learn 1.8.0 (via uv)	Logistic regression, ridge regression, random forest
Graph computation	networkx 3.6.1 (via uv)	PageRank, betweenness centrality, degree tracking
Web search	Brave Search API	Gap-fill agent web search

Server	macOS, Apple Silicon M1 16GB	Dedicated 24/7 server
Deployment	Vercel CLI v50+	Static HTML deployment from ~/openclaw/docs/scienceclaw/
Process supervision	macOS launchd	8 Thread 5 jobs registered in OpenClaw cron (jobs.json)

9.1 File Layout

Table 9. Key file paths.

Path	Content
~/openclaw/market-kg/	All Thread 5 Python scripts (22 scripts)
~/openclaw/market-kg/instructions/	Agent instruction files (edge-discovery.md, gap-identify.md, gap-fill.md, model-review.m
~/openclaw-knowledge/oncology-market/raw/{vertical}/{date}/	Raw API responses per vertical
~/openclaw-knowledge/oncology-market/staging/	Tier 1 KG-ready nodes + edges per vertical
~/openclaw-knowledge/oncology-market/kg/market_kg.json	Canonical KG output (single writer)
~/openclaw-knowledge/oncology-market/predictors/	edges_of_interest.json, gap_report.json, prioritised_gaps.json, gap_fill_proposals.json
~/openclaw-knowledge/oncology-market/manifests/	Weekly run manifests
~/openclaw-knowledge/oncology-market/MEMORY-HOT.md	Agent run log (appended by all four agents)
~/openclaw/docs/scienceclaw/market-kg.html	KG visualisation (Vercel)
~/openclaw/docs/scienceclaw/market-predictions.html	Predictions table (Vercel)
~/openclaw/docs/scienceclaw/scienceclaw-market.html	Landing page (Vercel)

9.2 Inter-Thread Data Flow

Thread 5 consumes two inputs from other threads. Thread 2 findings.json provides USEFUL-labelled trial anomaly signals as Tier 1 ANOMALY_SIGNAL edges. Thread 4 kg.json provides drug→indication edges as Tier 1 cross-references. Neither input requires re-querying the source APIs — Thread 5 reads the files directly. Thread 5 does not write back to Thread 2 or Thread 4.

10. Limitations

Manufacturing and distribution coverage. openFDA drug enforcement confirms manufacturer existence and recall history but provides no production volumes, contract manufacturing relationships, or supply chain flow data. HRSA 340B identifies covered entity locations but not drug dispensing volumes. Both verticals produce structural-only nodes.

Sales data currency. CMS Part D spending data has a ~2-year publication lag and covers Medicare-reimbursed spend only. It does not represent total US oncology market revenue. The revenue trajectory model is explicitly labelled as a directional index throughout the output and should not be used for revenue forecasting.

In-sample labels. All prediction model training labels are constructed from the KG itself. The approval model's post-2018 split is a reasonable proxy for next-generation vs legacy drugs but is not an externally validated outcome measure. CV AUC is indicative; true generalisation performance requires held-out data not available from free public sources.

Lead-lag data sparsity. The $r > 0.5$, $n \geq 8$ threshold for lead-lag correlation correctly eliminates most results in early deployment weeks. The predictor registry grows as weekly collection cycles accumulate annual data points. Meaningful lead-lag results require approximately 8+ weeks of weekly collection.

GLM-5 gap-fill reliability. The Gap Fill agent evaluates web search results for credibility and assigns relevance scores. The deterministic validation step (confidence = source_quality × relevance_score, minimum 0.45) provides a hard floor but does not verify that extracted numeric values are factually correct. Tier 2 proposals should be treated as directional signals, not confirmed data.

Memory pressure. Running Thread 5 alongside Threads 2, 3, and 4 on M1 16GB requires careful scheduling. The Thread 5 collection phase (06:00–09:30 UK) completes before Thread 2 starts (11:45 UK). The orchestrator checks for active Thread 2/3 processes and delays 30 minutes if found. The sklearn models are constrained to RF max 50 trees, depth 4, and n_jobs=1 to limit memory during concurrent operation.

Competitive coverage gap. Senior market research analysts in large pharmaceutical companies typically use IQVIA, Evaluate, or GlobalData. These platforms provide comprehensive commercial data coverage (total revenue, prescription volumes, ex-US markets) that cannot be replicated from free public sources. Thread 5's defensible value is cross-vertical signal integration, novelty detection (first-in entries, policy-drug confluences), and anomaly-oriented analysis — not comprehensive market sizing.

11. Discussion

Thread 5 represents a different intelligence layer from Threads 2 and 3. Where Thread 2 detects anomalies in clinical trial design and Thread 3 synthesises their market significance, Thread 5 maintains a persistent cross-vertical structural map of the US oncology market and surfaces patterns that are only visible when multiple verticals are examined simultaneously. The COMPETES_WITH and DOMINANT_IN Tier 3 edges, for example, are not discoverable from any single data source — they emerge from combining FDA Drugs@FDA sponsor data with ChEMBL indication linkages.

The OpenClaw agent architecture is central to Thread 5 in a way that differs from Threads 2 and 3. In Thread 2, the investigation agent reasons about specific trials. In Thread 5, the agents reason about the KG's structure itself — which patterns are meaningful, which data gaps most impair analytical value, whether the models are capturing the right signals. This is a higher-order reasoning task that benefits from the agent's ability to read multiple files, form hypotheses, and act on them by directing web searches.

A contrary viewpoint deserves explicit acknowledgment: a senior oncology market research analyst with access to IQVIA, Evaluate, and Bloomberg would likely produce higher-quality market intelligence. The system's advantage is continuous integration across heterogeneous public data sources, autonomous novelty detection, and directional signal generation at zero marginal cost per additional indication. Its disadvantage is that it cannot produce the revenue forecasts, market share estimates, or ex-US coverage that commercial platforms provide.

12. Illustrative Signal Scenarios

The following scenarios illustrate the types of signals Thread 5 is designed to surface once multiple weekly collection cycles have accumulated data.

Scenario A: Cross-vertical confluence in AML. NIH Reporter shows a 40% year-on-year increase in AML-targeted grant awards (academic vertical). ChEMBL shows three new compounds entering Phase 2 (drug discovery vertical). Thread 2 flags a first-in-class sponsor entry (trials vertical). The KG diff shows rising-degree nodes connecting all three. The Edge Discovery agent flags this as a criterion-A signal

(multi-vertical co-movement) and the Gap Fill agent retrieves the CMS AML spending trajectory to complete the prediction feature vector.

Scenario B: Policy-drug confluence. A new bill in the Senate Judiciary Committee targets a specific drug class for Medicare price negotiation. The Federal Register API surfaces the bill. The Tier 3 POLICY_RISK_FOR rule links it to affected Drug nodes. The Model Review agent notes that the disruption risk score for these drugs is elevated but the policy_risk_score feature is globally null — and raises this as a feature improvement proposal.

Scenario C: Supply risk signal. Three Class I recalls are filed for a chemotherapy agent within 60 days. The SUPPLY_RISK_FOR Tier 3 rule links the recall events to the indication. The shortage_risk_score feature becomes non-zero for affected drug×indication pairs, elevating their disruption risk score. The Edge Discovery agent flags the manufacturing→indication cross-vertical bridge as a criterion-B signal.

13. Data and Code Availability

All data was retrieved from publicly available APIs. No proprietary data sources were used. The system is implemented using the OpenClaw multi-agent framework with GLM-5:cloud as the analysis model and Kimi K2.5 for verification. scikit-learn 1.8.0 is used for prediction models. The implementation guide, agent instruction files, and configuration templates are available from the corresponding author on request. The demo instance is deployed at scienceclaw.ai/scienceclaw-market.html.

References

1. Singh R. Baseline-Deviation Analysis for Automated Detection of Anomalous Design Choices in Oncology Clinical Trials. v0.4 (March 2026). [Thread 2 companion paper.]

2. Singh R. Autonomous Market Intelligence for Science R&D;: A Multi-Agent Configurable Web Service with Cross-Model Verification. v1.1 (March 2026). [Thread 3 companion paper.]

3. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine Learning in Python. J Mach Learn Res. 2011;12:2825–2830.

4. Hagberg AA, Schult DA, Swart PJ. Exploring Network Structure, Dynamics, and Function using NetworkX. Proceedings of the 7th Python in Science Conference. 2008:11–15.

5. Lundberg SM, Lee SI. A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems. 2017;30.

6. Brandes U. A faster algorithm for betweenness centrality. J Math Sociol. 2001;25(2):163–177.

7. Page L, Brin S, Motwani R, Winograd T. The PageRank Citation Ranking: Bringing Order to the Web. Stanford InfoLab Technical Report. 1999.

8. Mendez D, Gaulton A, Bento AP, et al. ChEMBL: towards direct deposition of bioassay data. Nucleic Acids Res. 2019;47(D1):D930–D940.

9. Ochoa D, Hercules A, et al. The next-generation Open Targets Platform. Nucleic Acids Res. 2023;51(D1):D1353–D1359.

10. Zarin DA, Tse T, Williams RJ, Carr S. Trial Reporting in ClinicalTrials.gov — The Final Rule. N Engl J Med. 2016;375(20):1998–2004.