# To Be Updated Sections

* Phase 1 data extraction can use more and better sources
* News\_fetcher.py Newsapi endpoint can be changed from headlines to everything for more info
  + And this is free tier, we need better methods for news extraction
* Extractor.py can use llms instead of beautifulsoup
* Dedicated section for Academic papers
* Second run iteration for deeper research
* Actor input for large quantities
* Agents for specific professions and analyses
* Better evaluation
* Contact anthropic for financial service

1. High-level pipeline (merged)

Input: list of company names (CSV/DB).

Canonicalization / Identity resolution: fuzzy-match names → canonical name, ticker, CIK, website, name variants. (Keep your CompanyIdentifier module.)

Discovery (API-first):

Query authoritative structured sources first: ClinicalTrials.gov, openFDA, EDGAR, PubMed, USPTO/Google Patents.

Run SERP / News APIs (SerpApi / Scrapfly / NewsAPI / GDELT) to discover press releases, coverage, partnerships, and funding announcements.

Use targeted web scraping (company press pages, investor relations) as fallback for missing/blocked items.

Fetcher & Archive: fetch API JSON and discovered URLs; archive raw HTML/JSON to object storage (S3). Use lightweight HTTP fetching where possible; Selenium only for pages requiring JS or for manual QA.

Extractor & Normalizer:

For structured APIs: deterministic parsers (no LLM).

For unstructured pages: LLM-based extractor (instruction-tuned, deterministic prompts) + rule-based validators.

Must return standardized JSON + per-field provenance, evidence\_snippet (≤25 words), extraction\_confidence.

Structured Store: persist parsed profiles (Postgres + Elasticsearch) and raw docs (S3). Also store vector embeddings optional for semantic search.

Feature Engineering & Valuation: compute trial\_counts\_by\_phase, latest\_funding, cash\_runway, pipeline\_NPV (use your DrugPipelineValuator), risk\_scores (use your RiskAssessmentEngine).

Scoring & Confidence: apply weighted scoring model with confidence metric derived from coverage & source quality.

Reporting & Visualization: company dashboards, portfolio heatmaps, risk-vs-return scatter plots, and downloadable reports (PDF/CSV).

Validation & Tuning: compare to client reference set (Spearman ρ, precision@k), run provenance audits, and iterate queries/weights.

2. Data sources & discovery strategy (merged)

Canonical structured sources (priority)

ClinicalTrials.gov API (trial records, NCT IDs) — first call for pipeline data.

openFDA / Drugs@FDA (approvals, safety events).

EDGAR / SEC filings (public company financials / 10-K/10-Q / 8-Ks).

PubMed / Crossref (peer-reviewed publications).

USPTO / Google Patents (IP signals).

Discovery & news

SERP / News APIs (SerpApi, NewsAPI, GDELT) to gather candidate URLs and headlines for each company variant.

Industry outlets (FierceBiotech, BioPharmaDive, Stat, Endpoints) — sourced via SERP API results.

Targeted scraping (fallback)

Company press/IR pages, investor decks, local news archives; use HTTP fetchers + BeautifulSoup where possible, Selenium only for pages that need JS or interactive loading.

Paid vendors (optional, recommended for parity)

Crunchbase / PitchBook / S&P Global — best for private funding and cap table details; if client wants parity, budget for these.

3. Merged system architecture (components)

API connector layer: clients for ClinicalTrials.gov, openFDA, EDGAR, PubMed, Patent queries.

Discovery layer: SERP & News API client; query templates generator (uses name variants).

Fetcher & Archive: HTTP fetcher with proxy fallback (ScrapingBee/Bright Data), S3 raw archive, basic HTML sanitizer.

Scraper module: your PharmaWebScraper adapted to be a fallback component — search via SERP first, then fetch/pr-scrape. Limit Selenium usage.

Extractor:

Rule-based parsers for structured JSON (ClinicalTrials.gov, EDGAR).

LLM Extractor microservice for unstructured pages: strict JSON output + evidence\_snippet + extraction\_confidence. Use low temperature.

Processing / Feature engine: your DataProcessor, DrugPipelineValuator, FinancialHealthAnalyzer, RiskAssessmentEngine — plug them into derived metrics pipeline.

Storage & Indexing: Postgres for normalized records, Elasticsearch for search, S3 for raw docs, Vector DB optional (Pinecone/Weaviate).

Orchestration & Workers: Celery / Kubernetes jobs to run company jobs in parallel with rate-limit queues.

Validation & QA: validation module to compute Spearman ρ, precision@k, coverage metrics and provenance audits.

API & Dashboard: FastAPI + React/Streamlit dashboard, downloadable reports, provenance drill-down links.

4. LLM guidance (merged)

Role: LLMs act only as extractors/normalizers/summarizers — never as the primary retriever.

Model choices:

Hosted: OpenAI or Anthropic for high-quality extraction/synthesis.

Local: Ollama-managed instruction-tuned models (Llama2-Chat / Mistral-Instruct) if on-prem required. Do not use DialoGPT for structured extraction.

Prompt rules:

Strict JSON-only output, with schema enforced.

For each extracted field return value, evidence\_snippet (copy ≤25 words from page), source\_url, fetched\_at, and extraction\_confidence (0–1).

If no evidence, return null and no\_evidence\_found:true.

Use low temperature (0–0.2), system role instructing to avoid inventing facts.

5. Schema (canonical JSON example)

{

"company\_name": "Acme Pharma Inc.",

"canonical": {"resolved\_name":"Acme Pharmaceuticals, Inc.","website":"https://acme.com","ticker":"ACME"},

"financial": {"public":true,"latest\_10k\_date":"2024-12-31","revenue":12000000,"cash\_on\_hand":25000000,"last\_funding":{"round":"Series C","amount":35000000,"date":"2023-09-01"},"provenance":[{"source\_url":"...","source\_type":"edgar","fetched":"2025-09-20","evidence\_snippet":"..."}]},

"pipeline": {"candidates":[{"name":"ACM-101","phase":"Phase II","trials":[{"nct":"NCT0123456","status":"Recruiting","source\_url":"...","fetched":"2025-09-20","evidence\_snippet":"ACM-101 is currently in Phase II trials"}]}],"counts\_by\_phase":{"phase1":1,"phase2":1}},

"regulatory": {"fast\_track":true,"submissions":[{"type":"IND","date":"2024-03-10","status":"active","source\_url":"...","evidence\_snippet":"IND filed"}]},

"derived\_metrics": {"trial\_score":72,"financial\_score":55},

"final\_score": {"value":62.5,"confidence":0.78},

"raw\_documents": [{"source\_url":"...","fetched":"...","raw\_text\_snippet":"..."}]

}

6. Scoring & confidence (merged)

Default weights (tunable): pipeline 40%, financial 30%, partnerships 20%, sentiment 10%.

Confidence calculation: confidence = 0.6 \* structured\_coverage + 0.3 \* mean\_source\_quality + 0.1 \* corroboration\_index.

Source quality mapping: ClinicalTrials.gov/openFDA/EDGAR = 1.0; Company IR/press = 0.8; Reputable industry press = 0.6; Aggregated news/blogs = 0.4.

7. Validation & parity testing

Metrics: coverage %, Spearman ρ, precision@k (k=10/20), per-field extraction accuracy (manual audit of sample), false positive/negative analysis.

Workflow: run on client-provided reference set (e.g., 50–100 companies), compute metrics, then tune SERP templates, source weights, and extraction prompts iteratively.

8. Operational notes & scale

Rate limits: prefer APIs; use SERP paid plans for discovery. Cache results for TTL: clinical/regulatory daily, news weekly.

Selenium usage: avoid large-scale Selenium; use it only for small fraction of pages needing JS.

Legal: respect ToS; suggest licensing Crunchbase/PitchBook if client demands private deal completeness.

Logging & audit: every data point must include provenance link for client review.

9. Migration & rollout (phased)

Phase A — Prototype: implement end-to-end for 20 companies (mix public/private). Use structured API + SERP + LLM extraction.

Phase B — Validation: tune & compare vs client reference set (Spearman/precision@k). Adjust weights.

Phase C — Scale: run on 600 companies in batches, monitor failures, add retries and proxy scraping fallbacks.

Phase D — Handoff: deliver dashboards, docs, and training materials.

**10.1 Automated Investment Intelligence Engine**

**Catalyst Identification Module:** Parse clinical trial timelines from ClinicalTrials.gov API responses to automatically calculate expected readout dates within 6-12 months based on enrollment status and study duration fields. Cross-reference FDA calendar RSS feeds with company pipeline stages to estimate regulatory milestone timing using deterministic date calculations. Flag upcoming patent expirations by parsing USPTO data for key compounds and calculating expiration dates. Weight each catalyst using predefined impact scores based on development phase and indication market size.

**Peer Benchmarking Engine:** Generate relative rankings by calculating percentile scores for each company's metrics within their therapeutic area classification. Use the normalized company profiles to automatically compute sector medians for pipeline\_NPV, cash\_runway, and trial\_success\_rates. Flag statistical outliers (companies >2 standard deviations from peer group means) as potential opportunities or risks. Generate relative valuation scores using z-score normalization against peer groups.

**Scenario Modeling Framework:** Build automated probabilistic models using the existing DrugPipelineValuator with Monte Carlo simulation loops. Generate base/bull/bear scenarios by adjusting probability of success rates ±20% from baseline assumptions. Calculate portfolio-level risk using correlation matrices derived from therapeutic area classifications. Model downside scenarios by stress-testing cash runway against extended development timelines.

**10.2 Automated Recommendation Generator**

Transform quantitative scores into recommendation tiers using rule-based decision trees with predefined thresholds:

**Scoring Integration Rules:**

* Strong Buy: final\_score >80 AND confidence >0.75 AND catalyst\_count >2
* Buy: final\_score 65-80 AND confidence >0.6 AND risk\_flags <2
* Hold: final\_score 45-65 OR mixed threshold conditions
* Sell: final\_score <45 OR confidence <0.4 OR critical\_risk\_flags >0

**Automated Risk Flag Detection:** Implement threshold-based alerts for cash runway <4 quarters, clinical hold status detected in trials data, competitive programs in Phase III identified through ClinicalTrials.gov searches, or negative sentiment scores from news aggregation exceeding -0.6.

**10.3 Automated Evaluation Methodology**

**Performance Tracking System:** Store recommendation snapshots with timestamps and track subsequent 6-month stock performance data via financial APIs. Calculate automated hit rates by comparing predicted performance categories against actual returns. Generate monthly performance reports showing recommendation accuracy by confidence level and therapeutic area.

**Data Quality Monitoring:** Implement automated coverage metrics calculating percentage of fields populated per company profile. Set data freshness alerts when structured sources haven't been updated beyond TTL thresholds. Monitor extraction confidence distributions to flag potential LLM degradation or source quality issues.

**Statistical Validation Framework:** Calculate rank correlations between model scores and available analyst consensus ratings (where public). Measure prediction stability by tracking how often company recommendations change between monthly refreshes. Flag companies with high score volatility for manual review.

**10.4 Automated Insight Generation**

**Executive Summary Automation:** Generate standardized investment theses using template-based text generation with score insertions: "Company X shows [Strong/Moderate/Weak] investment potential with pipeline NPV of $[value]M and [number] near-term catalysts. Primary risks include [top 2 risk factors]. Recommendation: [tier] with [confidence]% confidence."

**Alert System Architecture:** Implement threshold-based monitoring for >20% changes in final\_score, new clinical holds detected, or competitor program advances into later phases. Generate daily alert digests for portfolio monitoring with automated filtering by severity levels.

**Comparative Ranking Reports:** Generate automated monthly rankings of all 600 companies with percentile distributions across key metrics. Highlight top/bottom deciles using automated percentile calculations and flag score movements >1 quartile between periods.

**10.5 Feasible Operational Metrics**

**Coverage Quality Assessment:** Automated calculation of profile completeness percentages across the 600-company universe. Target metrics: >90% companies with recommendation scores, >75% with catalyst identification, >95% with peer rankings. Generate automated flags for companies below minimum data thresholds.

**System Performance Monitoring:** Track API response times, extraction processing delays, and data refresh cycle completion rates. Implement automated alerts for system components exceeding performance thresholds (>5 second API responses, >24 hour data staleness).

**Cost Efficiency Tracking:** Monitor API usage costs against data points extracted. Calculate cost-per-company metrics and flag expensive sources providing low-value data. Implement automatic source prioritization based on cost-effectiveness ratios.

**10.6 Automated Learning Framework**

**Score Calibration System:** Monthly automated recalibration of scoring weights using performance feedback data. Implement A/B testing framework randomly assigning scoring model variants to company subsets and measuring subsequent accuracy. Store model performance history for automated rollback capabilities.

**Anomaly Detection:** Implement statistical process control to identify companies showing unusual scoring patterns or data quality issues. Flag companies with scores >3 standard deviations from therapeutic area norms for review. Monitor extraction confidence trends to detect systematic LLM performance degradation.

**Feedback Integration Pipeline:** Automatically incorporate market feedback by tracking correlation between recommendation changes and subsequent stock movements. Use this data to automatically adjust confidence scoring models and threshold parameters through gradient descent optimization.

This automated intelligence layer provides feasible evaluation and insight generation capabilities without requiring human intervention, maintaining alignment with the existing API-first technical architecture while delivering actionable investment intelligence through programmatic analysis.