MAY 2026

RAGNEROCK

The AI Efficiency Paradox: Why Your Layoff Savings Are Disappearing Into Query Costs

A CFO's Guide to Controlling AI Infrastructure Spend Without Reversing Course

Executive Summary

A CFO's Guide to Controlling AI Infrastructure Spend Without Reversing Course

> In 2025, companies restructured workforces betting on AI-driven efficiency. By mid-2026, those savings are being consumed by AI infrastructure costs nobody projected accurately.
> The reason is structural: traditional AI charges full price for every query — including every repeat.

$0.125	$625	$19,010	$228,125	$240
per query (Claude Opus 4.7)	per day (5,000 queries)	per month	per year (pay-per-query AI)	per year (Ragnerock)

Abstract

Traditional AI systems charge per query. This turns what should be a one-time cost into a recurring expense that grows with every user interaction. The billing model that made sense for early pilots becomes a liability at scale: ten users at $1,000 a month becomes $100,000 a month at one hundred users, not because headcount grew, but because query volume compounds as adoption deepens and users trust the system more.

The waste embedded in this model is significant. Support teams answer the same questions daily. Sales teams pull the same customer data repeatedly. Analytics teams run the same reports on a weekly cycle. In a conventional AI architecture, each of those interactions triggers full LLM inference and pays full token costs, regardless of whether the answer already exists.

This whitepaper introduces Ragnerock. The core idea is a separation that most AI architectures skip entirely: extraction happens once, querying costs almost nothing. Ragnerock processes source data with AI and stores the results as structured, persistent artifacts queryable with standard SQL. No LLM runs at query time. Costs scale with data volume, not with how many times people ask questions.

In practice, this reduces per-query costs from $0.125 to less than $0.01. At the volumes a typical mid-size enterprise runs, that difference exceeds $200,000 a year.

Table of Contents

The AI Efficiency Paradox

A CFO's Guide to Controlling AI Infrastructure Spend Without Reversing Course

> In 2025, over 55,000 technology workers lost their jobs as companies bet on AI-driven efficiency.[1] CEOs announced bold visions where AI agents would handle customer support, automate analysis, and deliver the productivity gains that justified workforce reductions. Marc Benioff at Salesforce was among the most vocal, cutting 4,000 customer support roles and publicly stating that AI agents now handle roughly half of all support work.[2]

Six months later, CFOs are discovering a painful reality. The promised savings are being consumed by AI infrastructure costs that scale far faster than anyone projected. The severance packages hit the P&L in Q1 2025. The new AI infrastructure started ramping in Q2. By Q3 and Q4, usage patterns stabilized. Now, in mid-2026, the full run-rate costs are visible, and they're uncomfortable.

> **The Core Problem: Traditional AI Systems Charge Per Query.**
>
> Every time your team asks the same question, runs the same analysis, or processes similar data, you pay full price for the same computation. This per-query pricing model, standard across major cloud providers and AI platforms, turns what should be a one-time extraction cost into a recurring expense that compounds with every user interaction.

The financial impact is substantial. **Companies processing thousands of queries daily on traditional AI infrastructure are seeing costs in the millions annually, with significant portions wasted on repeat queries.** This article explains why AI costs are spiraling, how query-based pricing creates hidden waste, and how companies are solving this without abandoning their AI investments or admitting the initial strategy was flawed.

The Year AI Became Expensive

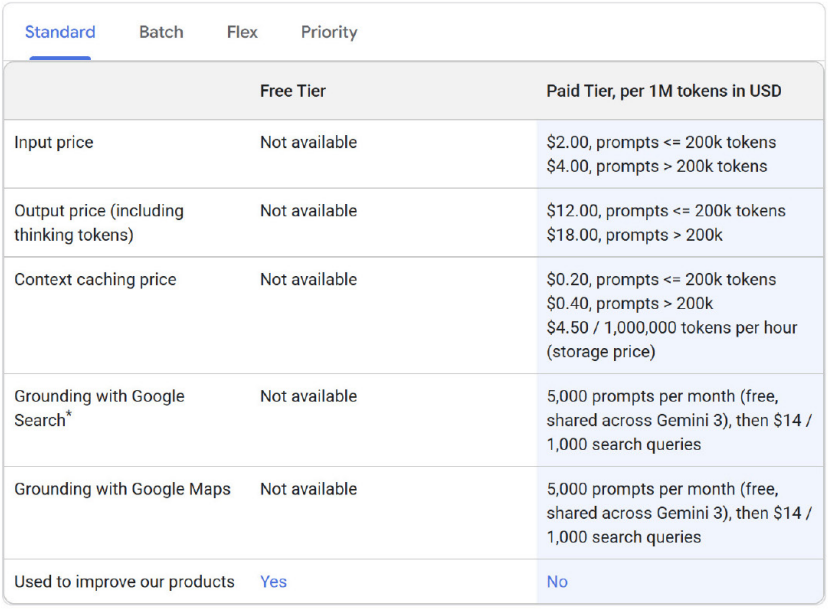
2026 marks the year AI infrastructure shifted from growth economics to revenue economics. Major model providers have raised token API costs, reduced quotas on monthly plans, experimented with restricting features, throttled traffic during peak usage, and more generally pivoted from customer acquisition to monetization. If your business has bet big on AI, understanding how to navigate this transition is essential. The companies that succeed won't be those that use the most sophisticated models. They'll be the ones that understand how their usage patterns map to provider billing models and architect accordingly.

How AI Billing Actually Works

AI cost optimization starts with understanding how your organization's usage patterns map to your provider's billing model. There are two main approaches: consumption-based pricing and fixed quotas.

Consumption pricing is metered in tokens, with separate rates for input tokens that you send to the AI and output tokens representing the responses you receive.[3,4] Gemini 3.1 Pro charges $2.00 per million input tokens and $12.00 per million output.

Gemini 3.1 Pro — $2.00 per million input tokens · $12.00 per million output tokens



Standard | Batch | Flex | Priority

	Free Tier	Paid Tier, per 1M tokens in USD
Input price	Not available	$2.00, prompts <= 200k tokens $4.00, prompts > 200k tokens
Output price (including thinking tokens)	Not available	$12.00, prompts <= 200k tokens $18.00, prompts > 200k
Context caching price	Not available	$0.20, prompts <= 200k tokens $0.40, prompts > 200k $4.50 / 1,000,000 tokens per hour (storage price)
Grounding with Google Search*	Not available	5,000 prompts per month (free, shared across Gemini 3), then $14 / 1,000 search queries
Grounding with Google Maps	Not available	5,000 prompts per month (free, shared across Gemini 3), then $14 / 1,000 search queries
Used to improve our products	Yes	No

Claude Opus 4.7 — $5.00 per million input · $25.00 per million output

MODEL	BASE INPUT	5M CACHE WRITE	1H CACHE WRITE	CACHE HITS	OUTPUT
Claude Opus 4.7	$5/MTok	$6.25/MTok	$10/MTok	$0.50/MTok	$25/MTok
Claude Opus 4.6	$5/MTok	$6.25/MTok	$10/MTok	$0.50/MTok	$25/MTok
Claude Opus 4.5	$5/MTok	$6.25/MTok	$10/MTok	$0.50/MTok	$25/MTok
Claude Opus 4.1	$15/MTok	$18.75/MTok	$30/MTok	$1.50/MTok	$75/MTok
Claude Opus 4	$15/MTok	$18.75/MTok	$30/MTok	$1.50/MTok	$75/MTok
Claude Sonnet 4.6	$3/MTok	$3.75/MTok	$6/MTok	$0.30/MTok	$15/MTok
Claude Sonnet 4.5	$3/MTok	$3.75/MTok	$6/MTok	$0.30/MTok	$15/MTok
Claude Sonnet 4	$3/MTok	$3.75/MTok	$6/MTok	$0.30/MTok	$15/MTok
Claude Haiku 4.5	$1/MTok	$1.25/MTok	$2/MTok	$0.10/MTok	$5/MTok
Claude Haiku 3.5	$0.80/MTok	$1/MTok	$1.6/MTok	$0.08/MTok	$4/MTok
Claude Opus 3 (deprecated)	$15/MTok	$18.75/MTok	$30/MTok	$1.50/MTok	$75/MTok
Claude Haiku 3	$0.25/MTok	$0.30/MTok	$0.50/MTok	$0.03/MTok	$1.25/MTok

Claude Opus 4.7 charges $5.00 per million input and $25.00 per million output. Input tokens are significantly cheaper than output tokens. One million tokens is roughly 750,000 words. This pricing disparity exists because techniques like KV caching make the "read" stage of token generation computationally cheaper than the "write" stage. Modern chain-of-thought reasoning further adds to the computational cost of generating output tokens. As a result, your AI costs in a consumption billing model will be largely a function of how much data you feed into your AI and, more importantly, how much output you generate from it.

> **Fixed-Quota Plans: A Warning**
>
> Fixed-quota plans, such as Claude Max, provide a fixed quantity of input and output tokens you can consume over some time interval. Essentially, you're purchasing "up to X" tokens per month. However, the exact quota is typically not published and varies dynamically with system load. **This makes fixed-quota billing less reliable than consumption billing because you can get rate-limited, often at the worst possible time during heavy workloads at peak business hours.** It is worth noting that fixed-quota billing is targeted at retail consumers; enterprise users aren't supposed to be on Claude Pro or Claude Max. For instance the enterprise plan is $20/month + API usage at the pay-as-you-go rate, it's not a quota-based system.

Why AI Initiatives Are Victims of Their Own Success

The tricky part in forecasting AI costs is that successful AI initiatives grow faster than linear projections suggest. As a project demonstrates value and gets adopted throughout the organization, usage increases and begets new use cases. What started as a pilot with ten users becomes an enterprise-wide tool with hundreds of users, each generating more queries than the pilot participants did.

Consider the simple example of an analytics chatbot. A user makes a query like "which products sold the most inventory last quarter?" In a naive implementation, each time any user asks that query, the chatbot application makes a call to the AI provider's API and consumes tokens. The more users asking questions, the more tokens consumed, even if questions are repeated or near-duplicates. This problem is especially prevalent in larger organizations where multiple users may be making essentially the same requests throughout the day.

The Hidden Costs in Three Layers

The reality of AI costs breaks down into three layers that interact in unexpected ways.

Layer 1: Initial Extraction & Processing
The first layer is initial extraction and processing. Every time data enters your system, whether it's a customer support ticket, a legal document, or a sales inquiry, AI must process it. This involves breaking documents into chunks called embeddings, running inference through large language models, and generating structured outputs. The cost per operation varies depending on complexity, but it represents the foundation of all AI work.

Layer 2: Storage in Specialized Vector Databases
The second layer is storage in specialized vector databases. Processed data gets stored for retrieval, and monthly costs include storage volume measured per gigabyte, compute resources for searches, and performance tiers based on speed and latency requirements. Industry pricing runs from $70 to $500 per month for mid-size deployments, scaling to $2,000 to $10,000 monthly at enterprise scale.[5] These costs are predictable and manageable.

Layer 3: Retrieval & Re-Inference: Where Costs Explode
The third layer is where costs explode: retrieval and re-inference happens every single time someone queries the system. A user asks something straightforward like "What are the top reasons customers churn?" The system retrieves relevant data from the vector database, incurring compute costs. It sends that data plus the query to the LLM, paying token costs based on input and output length. It generates a response, incurring inference costs. Each query costs anywhere from five cents to fifty cents for typical operations, and one to five dollars for complex analysis.

The Cost Math & Why Projections Were Off

Consider a mid-size deployment supporting customer support AI. Five thousand support tickets come in daily. 60% are repeat question patterns. The average query involves 20,000 input tokens and 1,000 output tokens. Using Claude Opus 4.7 at $5.00 per million input tokens and $25.00 per million output tokens, that works out to $0.125 per query.

The daily cost calculation is straightforward but sobering:

Daily Cost — Traditional Approach	
5,000 queries × 21,000 tokens	$105 million tokens per day
Daily cost at $0.125 per query	$625.00 daily
Monthly cost	$19,010 monthly
Annual cost	over $228,125 annually
Waste from 60% repeat queries alone (3,000 queries/day)	$375 daily · $136,875 annually

This is just one use case. Multiply across sales, analytics, compliance, and operations, and enterprises are looking at millions in inference costs, with substantial percentages wasted on processing data they've already analyzed.

Why Your Cost Projections Were Off

The initial AI cost projections failed for three specific reasons, each rooted in assumptions that seemed reasonable but proved incorrect.

- **Assumption 1: AI costs scale linearly.** If ten users cost $1,000 per month, surely one hundred users cost $100,000 per month. What actually happens is that costs scale with query volume, not user count. As adoption increases, users trust the system more and submit more queries per person. They push boundaries with more complex queries. They iterate on answers with exploratory analysis. The result is exponential growth. Companies typically see query volume increase three to five times within the first six months of deployment as users become comfortable with AI tools.[6]
- **Assumption 2: Efficiency would offset costs.** Leadership believed that because AI handles ten times the work of human employees, even higher costs would be justified. What actually happens is that AI doesn't replace work so much as change how work is done. Instead of humans searching for answers, they ask AI multiple questions to refine results. They validate AI outputs, often requiring more queries. They explore tangential questions they wouldn't have pursued manually because the friction of asking was too high before. This isn't a failure of AI. It's how knowledge work actually functions when you remove barriers to asking questions. But it means query volumes far exceed initial projections.
- **Assumption 3: Cloud vendors would compete on price as AI became commoditized.** Token pricing has remained relatively stable instead. Major providers including OpenAI, Anthropic, Google, and AWS maintain similar pricing structures. Models at the GPT-4 tier cost $10 to $30 per million tokens. Mid-tier models run $3 to $10 per million tokens. Embedding models cost $0.10 to $0.30 per million tokens.[6] While these rates have decreased from early 2024 peaks, they're not dropping fast enough to offset volume growth.

> **Board-Level Questions CFOs Are Facing in Q2 2026**
>
> Boards are scrutinizing AI investments with increasingly uncomfortable questions. Where's the promised ROI from restructuring? Why are cloud bills increasing while headcount is down? Can we quantify the productivity gains? What's our path to profitability with these costs? CFOs need answers that don't involve admitting the AI bet was premature.

The Solution: Encode Interaction Patterns Into Workflows

For a cost-conscious organization, the solution is to identify repetitive interaction patterns and encode them into agentic workflows. Then the request runs once rather than on every user interaction, and the results are persisted for reuse.

Traditional AI systems conflate two distinct operations:

- The **first is extraction**, which is expensive and should happen once. This involves processing raw data, generating structured outputs, and storing results.
- The **second is querying**, which should be cheap and happens repeatedly. This involves searching structured data and returning results without needing LLM inference.

The problem is that traditional architectures run expensive LLM inference every time you query, even when the answer already exists in structured form. The alternative is to separate these operations entirely.

Consider our analytics chatbot example again. When a user asks "which products sold the most inventory last quarter," a traditional system calls the AI API every single time that question is asked. With an optimized approach, the request runs once as an agentic workflow. The results are persisted as strongly typed artifacts that can be integrated into downstream systems or queried directly with SQL. Crucially, tokens are only consumed once when the workflow runs, not every time a user asks that question.

This has performance advantages too. Executing a SQL query on a database is much faster and more scalable than running LLM inference on a frontier model, and it reuses your existing data infrastructure. The cost advantage is obvious: token consumption is reduced proportional to how frequently user interactions are repeated, which can represent a **10x or better cost advantage** depending on the organization and use case.

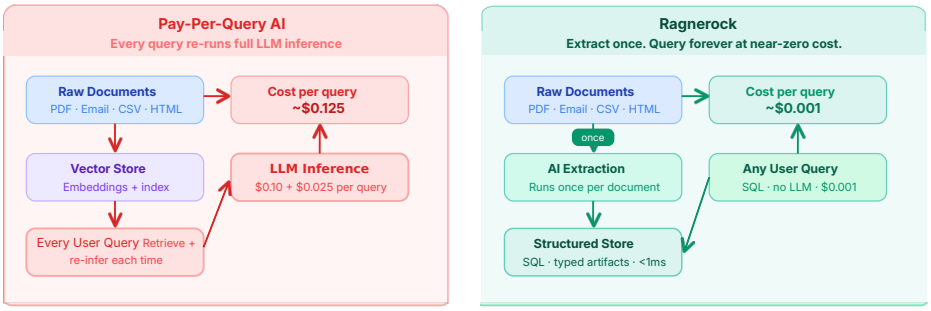


Figure 1 — Architecture comparison: Pay-Per-Query AI re-runs full LLM inference ($0.125/query) on every query. Ragnerock extracts once; all subsequent queries are SQL at ~$0.001 — a 125× cost reduction.

How Query-Optimized Architecture Works in Practice

Ragnerock's approach exemplifies this principle. Rather than running AI inference at query time, it creates queryable structured data when information first enters your system. The process has two distinct steps. Initial processing involves AI extraction. Raw documents arrive as PDFs, emails, tickets, tabular formats like CSV and excel workbooks or other formats. AI extracts key information exactly once. Results get stored as structured records in your existing data infrastructure. Query execution involves no AI inference. Users query via standard SQL or semantic search. The system searches structured tables at millisecond latency. It returns results directly from pre-processed data. There's no LLM call and no token charges. The architectural principle is straightforward. AI extraction runs when data enters the system. Results persist as structured records, queryable with standard SQL or semantic search at millisecond latency. No LLM runs at query time. Costs scale with data volume rather than query volume. This fundamentally changes the economics of operating AI systems at scale.

The cost comparison is stark. Using the traditional approach with query-time inference for customer support FAQ analysis, someone asks "What are the top 10 churn reasons?" The system retrieves relevant data from the vector database, incurring compute costs. It sends that data plus the query to the LLM, which at 20,000 input tokens times $5.00 per million costs $0.10.

It generates a response at 1,000 output tokens times $25.00 per million, adding $0.025. Total cost per query runs about $0.125. One hundred queries cost $12.50. A thousand queries monthly cost $125. Annually, that's $1,500.

Using a query-optimized approach, there's a one-time processing cost to extract churn reasons from all tickets and store them as a structured table in the existing data lake. Subsequent queries are SQL searches against the structured table, costing roughly one-tenth of a cent. One hundred queries cost ten cents. A thousand queries monthly cost $1.00. Annually, that's $12.00 plus the initial extraction cost.

> **Cost at scale (1,000 queries/month)**: Pay-Per-Query AI costs $1,500/year ($0.125/query × 12,000 queries). Ragnerock costs $12/year ($0.001/query × 12,000 queries) — a 125× reduction. See Figure 2.

What Actually Changes During Implementation

The beauty of query-optimized architecture lies in what you're not changing. You keep your existing AI models and providers. You keep your current data infrastructure. Your team's workflows and tools remain the same. Your compliance and security posture stays intact.

What you add is a structured data layer between raw inputs and queries, automated extraction when data enters the system, and standard SQL or semantic search over processed results.

Ragnerock integrates with existing infrastructure rather than replacing it. Data sources include SQL databases, Excel files, PDFs, HTML, images, and more. You bring your own AI provider keys from OpenAI, Anthropic, Google, or AWS. Outputs flow directly to your data warehouse, whether that's Snowflake, Databricks, or BigQuery. Source documents stay in your cloud storage on S3, Azure, or GCS.

> **The Architectural Principle: Nothing Leaves Your Infrastructure**
>
> Outputs flow directly to your data lake. Source documents stay in your cloud storage. Ragnerock adds the structured-data layer while everything else stays where it is.

Change Management Is Straightforward

Because most people see immediate benefits:

Engineering Teams

Face minimal changes to existing pipelines. Use standard SQL queries with familiar tools. Performance actually improves with millisecond latency compared to the seconds required for LLM responses.

Business Users

Experience the same or better user experience with faster results and no new tools to learn.

Finance

Gets predictable costs that scale with data rather than queries, clear ROI metrics, and board-ready reporting.

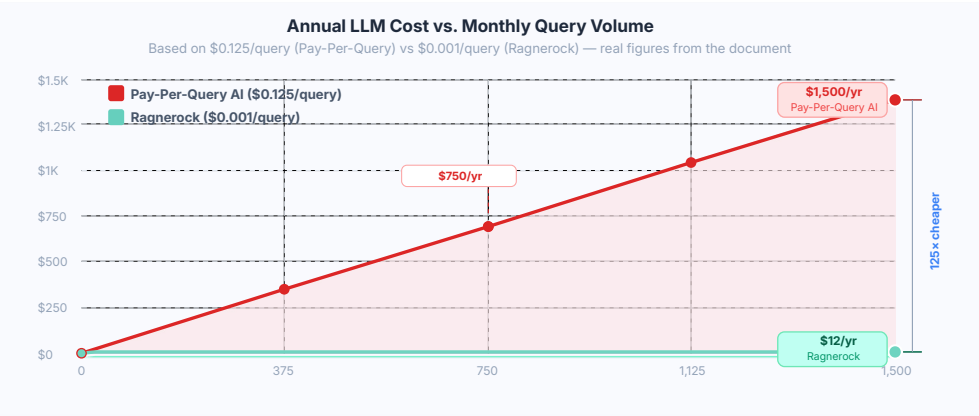


Figure 2 — Annual cost at scale: 1,000 queries/month costs $1,500/yr (PPQA) vs $12/yr (Ragnerock). Data from worked example in source document.

Don't Let Rising Costs Stop Your AI Journey

The AI transformation your company committed to in 2025 was strategic. The workforce restructuring was necessary. The bet on AI-driven efficiency was correct. The execution, however, needs refinement.

Traditional AI architectures treat every query as novel. They charge full price for repetitive work. They scale costs with adoption, creating a financial ceiling on the very efficiency they promise to deliver. This isn't a flaw in AI technology itself. It's a consequence of pricing models designed for different usage patterns than what actually emerges in production environments.

Query-optimized architecture completes the transformation by delivering on the original promise of sustainable AI economics that reward scale rather than punish it. By separating one-time extraction from recurring queries, by storing results as structured data, and by eliminating redundant inference costs, companies can achieve the cost structure their initial business cases assumed they were getting.

This isn't about abandoning AI or admitting mistakes. It's about demonstrating the financial sophistication to optimize as you scale. Your board expects you to show ROI on the AI investments that justified difficult workforce decisions. Query optimization gives you the numbers to do exactly that.

For CFOs Ready to Take Control of AI Costs

1. Understand how your usage patterns map to provider billing models
2. Identify repetitive interaction patterns in your organization
3. Encode those patterns into agentic workflows that run once and persist results
4. Monitor the cost reduction and operational improvements
5. Document everything for board reporting

> *"The companies that will succeed with AI in 2026 and beyond won't be those using the most advanced models. They'll be the ones that understand the economics deeply and architect accordingly."*

If you're interested in finding out how Ragnerock can help you scale your AI initiatives while keeping your token costs under control, schedule a brief call with us today or start a free pilot now.

Schedule a Brief Call → | **Start a Free Pilot →**

About Ragnerock

Ragnerock creates queryable data from any raw source and connects it to existing infrastructure. The platform applies AI extraction when data enters the system, storing results as structured records queryable via standard SQL or semantic search. By separating extraction from querying, Ragnerock eliminates recurring LLM inference costs while maintaining millisecond latency and complete audit trails. The outputs of agentic workflows in Ragnerock are durable, persisted, strongly typed artifacts that can be integrated into downstream systems or queried directly with SQL. Crucially, tokens are only consumed once when the workflow runs, not every time a user asks that question.

The platform is used by companies in financial services, healthcare, legal, and technology sectors where data volume is high, query patterns are repetitive, and cost efficiency is critical.

For more information, visit **www.ragnerock.com.** To discuss specific cost modeling for your organization, contact the Ragnerock team at **hello@ragnerock.com.**

Citations

1. Business Insider, "The List of Major Companies That Have Announced Layoffs So Far in 2025," https://www.businessinsider.com/list-companies-replacing-human-employees-with-ai-layoffs-workforce-reductions (accessed May 2026).
2. IndMoney, "Salesforce Layoffs: 4,000 Jobs Cut as CEO Marc Benioff Bets on AI Agents," https://www.indmoney.com/blog/us-stocks/salesforce-layoffs-4000-jobs-cut-as-ceo-marc-benioff-bets-on-ai-agents (September 2025).
3. https://platform.claude.com/docs/en/about-claude/pricing (May 2026)
4. https://ai.google.dev/gemini-api/docs/pricing#gemini-3.1-pro-preview (May 2026)
5. BuildMVPFast, "Pinecone vs Weaviate (2026): Vector Database Comparison," https://www.buildmvpfast.com/compare/pinecone-vs-weaviate (February 2026). Pricing ranges verified across Pinecone ($70/month for 10K users, scaling to $500-$2,000/month for 50K-100K users) and Weaviate platforms. Enterprise pricing confirmed via SpotSaaS and AlternativeTo pricing directories.
6. Industry pricing aggregated from OpenAI, Anthropic, Google (Gemini), and AWS Bedrock pricing pages. GPT-4 tier: OpenAI GPT-4 Turbo ($10/M input, $30/M output), Anthropic Claude Opus ($15/M input, $75/M output adjusted for context), Google Gemini Pro ($7/M input, $21/M output). Mid-tier: GPT-3.5 Turbo, Claude Sonnet, Gemini Flash. Embedding models: OpenAI text-embedding-3-large ($0.13/M), Google text-embedding-004 ($0.025/M). Pricing as of Q1 2026.

RAGNEROCK

RESEARCH INTELLIGENCE PLATFORM

Turn documents into databases.
Turn research into insight.
Turn weeks into minutes.

www.ragnerock.com