MCP SERVER | NO CODE | CLOUD HOSTED

Jina AI MCP

Ground your agent with real-time web data.

Jina AI (Search Foundation & LLM Grounding) provides your agent with real-time web intelligence and deep document context. It lets you extract clean text from any URL, perform semantic searches optimized for RAG, generate embeddings, and classify documents without needing to train a model.

A+ Quality Score 100/100

embeddings | rag | semantic-search | web-scraping | llm-grounding | data-extraction

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/jina-ai-search-foundation-llm-grounding

SCAN TO VIEW

The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect to the software your business already runs. We handle the hosting, the security, the credentials, the uptime — you get agents that actually do things.

We operate the world's largest managed MCP catalog. Major SaaS platforms, CRMs, databases, and cloud providers — running, monitored, production-ready. This MCP server is hosted and maintained by the Vinkius Cloud for AI Agents.

> *The agent doesn't manage credentials, doesn't manage uptime, doesn't manage security. Vinkius does.*
>
> — Architecture principle

Four Pillars of the Vinkius Runtime

01 — Security by design

Credentials stay encrypted at rest via AES-256. The AI agent never touches raw keys — they're injected into a sandboxed V8 isolate at runtime. Actions are logged, and connections have an emergency kill switch.

02 — Built on MCP Fusion

This MCP server was built with **MCP Fusion**, the open-source framework (Apache 2.0) that powers the entire Vinkius catalog. Schema-as-firewall strips undeclared fields, compiled PII redaction runs at zero overhead, and cryptographic lockfiles produce git-diffable audit trails.

03 — Deterministic observability

Eight immutable metrics per endpoint: request volume, p95 latency, error rate, active connections, cost attribution. A live payload feed logs every tool call with mutation detection.

04 — Autonomous operations

Servers are deployed, monitored, and patched autonomously. New capabilities and security patches ship weekly. Zero-downtime deployments ensure continuous availability across all managed MCP servers.

AES-256	Ed25519	24h TTL	V8 Isolate
Encryption at rest	PKI vault signatures	Ephemeral session keys	Sandboxed execution

One Token. Instant Access.

Every MCP server on Vinkius is accessed through a **Connection Token**. Tokens are generated in the cloud dashboard and produce a unique MCP endpoint URL. Paste this URL into any MCP-compatible client — no SDK required.

A single token can serve **multiple AI clients simultaneously**, or you can issue separate tokens per client for granular access control. Each token tracks its own request count, last activity timestamp, and can be individually enabled or revoked.

MCP ENDPOINT

```
https://edge.vinkius.com/{token}/mcp
```

Claude · Cursor · VS Code · Windsurf · Grok · Gemini

Security Is the Architecture

Security in Vinkius is not a feature — it's the foundation of the runtime. The gateway enforces multiple independent protection layers between AI agents and third-party APIs.

01 — Ed25519 PKI Vault

Every workspace has an Ed25519 Master Key. Session keys are generated ephemerally (24h TTL) and signed by the Master Key. Credentials never leave the vault boundary.

02 — V8 Isolate Sandboxing

Tool code runs inside isolated-vm V8 isolates with 64 MB memory caps and per-request timeouts. No filesystem access, no network access except through the SSRF-guarded fetch bridge.

03 — SSRF Guard

All outbound HTTP requests are DNS-resolved and validated before execution. Private IP ranges (10.x, 172.16-31.x, 192.168.x, AWS metadata 169.254.x) are blocked at the network layer.

04 — DLP & PII Redaction

A ResponseGuard pipeline intercepts every tool response. Configurable redaction patterns strip sensitive fields (emails, SSNs, card numbers) before data reaches the AI agent.

05 — Cryptographic Audit Trail

Every request is signed into a SHA-256 hash chain with Ed25519 signatures. Events form a tamper-proof, SIEM-exportable forensic record.

06 — Honeytoken Trap System

Phantom credentials are injected into isolated environments. If a honeytoken is used outside Vinkius infrastructure, the server is quarantined instantly.

Emergency Kill Switch

EU AI Act Art. 14(1) Compliant

The kill switch is an **emergency halt** mechanism — not a simple toggle. When triggered, it executes three actions atomically:

01 — Server deactivated

The MCP server is immediately taken offline across the entire cluster.

02 — All tokens revoked

Every connection token is invalidated. Total lockout — reconnection blocked until new tokens are issued.

03 — WebSocket connections killed

Active connections terminated via Redis pubsub broadcast. Propagates to every runtime node in the cluster.

Full Visibility. Zero Guesswork.

The Vinkius cloud dashboard includes a full MCP Governance suite — real-time analytics and security controls for production AI operations.

Control Plane

KPI dashboard with request volume, latency, success rate, token consumption, and AI-generated operational briefings.

Agent Activity

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

FinOps

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

Tool Health

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

Firewall & DLP

PII redaction activity, sensitive data protection counters, and security event timeline.

Incident Log

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

> Get started at **cloud.vinkius.com** — connect your AI agent in under 60 seconds.

Jina AI (Search Foundation & LLM Grounding) MCP

6 tools available
Cloud-hosted on Vinkius

If your agent needs to answer questions about the current state of the internet or specialized private documents, this MCP is how you connect it. You can strip away noise from live web pages using the reader tool, ensuring your client only gets clean, readable context for its answers. Beyond general search, you get structured, deep web results that are perfect for advanced RAG pipelines. Need to process huge PDF reports? Instead of feeding the whole thing at once, you segment the content into meaningful chunks and generate high-quality vector embeddings. You can even refine initial searches by running a precise reranking step against your query, making sure the most relevant pieces of information always surface first. Because Vinkius hosts this catalog, you connect to all these advanced search functions—from web scraping to classification—through one setup with any MCP-compatible client.

Core Capabilities

01 — Extracting clean content from live URLs

It pulls raw text from a website, stripping away navigation and clutter so your agent gets usable, readable information.

02 — Performing structured web searches

The service executes semantic web searches that return highly organized results built specifically for analysis by AI agents.

03 — Creating document vector embeddings

You convert raw text into high-quality numerical vectors, which power the ability to find similar documents across massive datasets.

04 — Improving search relevance with reranking

It reorders a set of potential search results based on how closely they match your specific query block, boosting accuracy.

05 — Categorizing text inputs (Zero-Shot)

You assign labels to text documents without having to train or build custom classification models first.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/jina-ai-search-foundation-llm-grounding** — connect your AI agent in three steps.

01 Subscribe to this MCP and provide your Jina AI API Key.

02 Connect the key to any MCP-compatible client (like Cursor or Claude).

03 Call a tool like `search_web_jina` to receive structured, context-rich web results.

The bottom line is you get reliable access to state-of-the-art search and data processing tools through one simple API key setup.

Built For

This MCP targets the developer who gets bogged down in manual data pipelines. If your agent needs more than just a simple database lookup—if it needs to read the web, process PDFs, or classify messy inputs—you need this.

AI Engineer

They use `segment_content` on large documents and then generate vector embeddings so their agents can query knowledge bases accurately.

Data Scientist

They test embedding models and reranking logic against real data without writing manual Python code or using cURL commands.

Automation Developer

They automate the extraction of clean web content from URLs and classify incoming documents for large-scale data ingestion pipelines.

What Changes When You Connect

01 You stop relying on outdated or internal knowledge bases. Using the `read_url_content` tool lets your client access fresh, live information directly from the web when it answers questions.

02 Instead of simple keyword matching, you perform a semantic search using `search_web_jina`. This ensures the results are context-rich and meaningful for complex agent reasoning.

03 Processing huge data files used to mean manual chunking. Now, use `segment_content` to break down long documents into semantically optimized chunks ready for RAG systems.

04 You don't need a machine learning team to label things. The `classify_texts` tool lets you categorize incoming data streams instantly using zero-shot techniques.

05 When initial search results are too noisy, the `rerank_documents` tool cleans up the list by reordering documents based on their true semantic match to your query.

Real-World Applications

Updating a company policy handbook

An agent needs to know the latest compliance rules. Instead of searching only internal docs, it calls `read_url_content` on the official government website and then uses `segment_content` to break the new rule into discrete chunks for accurate reporting.

Market research on a competitor

A data scientist wants to understand market sentiment. They run a semantic search using `search_web_jina` and then use `classify_texts` on the resulting articles to quickly count how many are positive, negative, or neutral.

Building a knowledge retrieval system

A developer needs to build an agent that answers questions about millions of pages. They first process those pages into vectors using `generate_embeddings`, and then use the vector index for fast, context-aware lookups.

Assessing document relevance

An initial search returns 50 articles on a topic, but only three are relevant to the specific sub-topic. The agent calls `rerank_documents` to automatically reorder and highlight the top three most pertinent sources.

Patterns to Avoid

Treating all data as uniform text

✗ AVOID

Sending an entire 10-page PDF into the agent's context window, causing it to miss key details due to token limits or noise.

✓ INSTEAD

First, use `segment_content` on the long document. Then, pass those smaller, semantically distinct chunks for retrieval and analysis.

Assuming search results are ordered by relevance

✗ AVOID

Relying on a default web search API that returns documents in mere listing order, forcing your agent to read through irrelevant material.

✓ INSTEAD

Always run the retrieved document IDs through `rerank_documents` to ensure the most contextually relevant data appears first.

Using simple keyword searches for complex topics

✗ AVOID

Asking an agent about 'Multi-head Latent Attention' and getting results that only mention the words, but miss the technical meaning.

✓ INSTEAD

Use `search_web_jina` to perform a semantic search. This finds articles based on conceptual similarity, not just word overlap.

The Right Fit

Use this MCP if your agent needs external data that changes over time or resides in diverse formats (URLs, large PDFs). If the job requires reading *current* web information or processing documents larger than a few paragraphs, you need its tools. Don't use it if all your required data is already neatly contained within a single, small database table; for that, a simple database connector is enough. However, if you are just classifying text based on existing labels (like 'Product' or 'Service'), the `classify_texts` tool handles this without needing to connect to a specialized ML service.

The Challenge of Context Overload

Today, when an agent needs to answer a question about a topic like 'Q3 market trends,' you have to manually check three places: the company blog (a URL), a 40-page PDF report, and maybe some structured data from another system. You end up copying key passages, pasting them into your prompt, and hoping the agent doesn't get confused by the sheer volume of text.

With this MCP, you simply point your client at the sources. The service handles the complexity: it strips noise using `read_url_content`, breaks down reports with `segment_content`, and organizes everything so your agent only processes clean, highly relevant chunks.

Jina AI (Search Foundation & LLM Grounding) MCP Provides Contextual Depth

You eliminate the need for manual web scraping scripts and complex data preparation. You don't have to write boilerplate code just to extract clean text from a URL or run basic semantic searches.

The difference is that your agent doesn't guess. It grounds its answers using structured, real-time intelligence pulled in through this MCP.

Jina AI Search & Grounding MCP - 6 Tools

These tools allow you to process text from the web, generate embeddings, reorder search results, classify documents, and chunk large files for advanced agent workflows.

#	TOOL	DESCRIPTION
01	`generate_embeddings`	Creates numerical vectors that represent the meaning of text, making it searchable by concept rather than just keywords.
02	`rerank_documents`	Takes a list of retrieved documents and reorders them to put the most relevant ones at the top based on your query.
03	`read_url_content`	Pulls clean, readable text content from any provided web address for direct use by an agent.
04	`search_web_jina`	Executes a semantic search across the web and returns structured data optimized specifically for AI analysis.
05	`classify_texts`	Assigns predefined categories to text inputs using zero-shot learning, without needing prior model training.
06	`segment_content`	Breaks down lengthy documents into smaller, semantically cohesive chunks suitable for vector storage and retrieval.

See It in Action

Real prompts you can use once this MCP is connected to your AI agent through Vinkius Cloud.

U Extract the main content from 'https://jina.ai/embeddings' as Markdown

Reading URL... I've extracted the clean text from the page. It covers Jina's embedding models (v2 and v3), multilingual support for 80+ languages, and common use cases like semantic search and recommendation systems. Would you like a summary?

U Search the web for the latest updates on 'DeepSeek-V3 architecture'

Searching the web... Based on the structured results from Jina Search, DeepSeek-V3 features a Multi-head Latent Attention (MLA) mechanism and a Mixture-of-Experts (MoE) architecture with 671B total parameters. Would you like me to read the full technical paper URL?

U Segment this long text into semantically cohesive chunks: [text content]

Segmenting content... I've broken your text into 5 semantically distinct segments. Each segment focuses on a single core topic (e.g., Introduction, Architecture, Benchmarks). This is now optimized for vector storage and RAG retrieval.

Frequently Asked Questions

01 **How does Jina AI (Search Foundation & LLM Grounding) MCP handle PDFs?**

You use the `segment_content` tool to break long documents into semantically meaningful chunks. This process optimizes the data for vector storage, ensuring your agent can retrieve specific passages instead of the whole file.

02 **Can Jina AI (Search Foundation & LLM Grounding) MCP search beyond my internal documents?**

Yes. The `search_web_jina` tool performs semantic web searches, giving your agent access to current information from the live internet.

03 **What is the difference between embeddings and simple text passing?**

Simple text passes raw words; generating vector embeddings (`generate_embeddings`) converts the meaning of the text into a numerical format, allowing your agent to find concepts that are similar but use different vocabulary.

04 **Does Jina AI (Search Foundation & LLM Grounding) MCP require me to train models?**

No. You can categorize new text using the `classify_texts` tool with zero-shot learning, meaning you assign labels without needing to build or fine-tune a specific model.

05 **How do I ensure my agent reads the most important parts of a webpage?**

Use the `read_url_content` tool first to extract clean text. Then, if necessary, use `rerank_documents` on search results to surface the highest-relevance sections.

Go Live in 60 Seconds

Get your connection token from **cloud.vinkius.com**, then paste the endpoint URL into any MCP-compatible client.

YOUR MCP ENDPOINT

```
https://edge.vinkius.com/[TOKEN]/mcp
```

CLIENT	WHERE TO CONFIGURE
Claude AI	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
Cursor	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
VS Code	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add `"jina-ai-search-foundation-llm-grounding": { "url": "…" }`
Windsurf	MCP Settings → `mcp_settings.json` → Add endpoint URL
ChatGPT	Settings → Tools & plugins → Add MCP server → Paste endpoint
Gemini	Extensions → Add MCP Server → Paste endpoint URL

ASK AN AI ABOUT THIS

Let your preferred AI explain this MCP server

- Ask ChatGPT
- Ask Claude
- Ask Perplexity
- Ask Gemini
- Ask Grok

READY TO CONNECT

Jina AI (Search Foundation & LLM Grounding) is live on Vinkius Cloud.

Get your connection token, paste it into your AI agent, and start building. No SDK. No deployment. Just results.

Start at cloud.vinkius.com →

vinkius.com · support@vinkius.com

INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by Jina AI (Search Foundation & LLM Grounding). All third-party trademarks, logos, and brand names are the property of their respective owners. Their use in this document is strictly for informational purposes to identify service compatibility and interoperability.

DOCUMENT INFORMATION

Generated	June 2026
MCP Server	Jina AI (Search Foundation & LLM Grounding) MCP
Server ID	`019d75bd-0f15-703c-a3f8-c6f0fc82246d`
Platform	Vinkius Cloud for AI Agents
Endpoint	`https://edge.vinkius.com/{token}/mcp`

LICENSE & USAGE

This document is generated automatically by the Vinkius PDF Engine. Content reflects the MCP server configuration at the time of generation and may change as updates are deployed. For the most current information, visit vinkius.com/mcp/jina-ai-search-foundation-llm-grounding.