AI-Powered Financial News Intelligence System

Problem & Solution

Fixing noisy, redundant financial news with an AI-first workflow

The Problem

- Redundant coverage across sources; hard to track unique stories
- Entity context is scattered (companies, sectors, regulators)
- Slow manual analysis; unclear stock impact
- Literal keyword search misses thematic links

Our Solution

- ✓ Intelligent deduplication using embeddings + cosine similarity
- ✓ High-precision entity extraction and classification
- ✓ Stock impact mapping with confidence levels
- ✓ Context-aware queries + sentiment & price impact prediction

≥95% Dedup Accuracy

Multi-agent AI

Groq Llama + FastAPI

Core Features at a Glance

Five capabilities powering real-time market intelligence

Intelligent Deduplication

Embeddings + cosine similarity consolidate duplicate articles into canonical stories.

≥95% accuracy · threshold ≥ 0.85

Entity Extraction & Classification

Companies, sectors, regulators, people, events with confidence scoring via Groq Llama.

≥90% precision · Companies 100% · Sectors 60–80%

Stock Impact Mapping

Maps news to affected tickers with confidence by impact type (direct/sector/regulatory).

Direct 100% · Sector 60–80% · Regulatory variable

Context-Aware Query System

Natural-language queries with semantic theme matching and smart context expansion.

Response < 5s · Semantic search ≥ 0.70

Sentiment & Price Impact

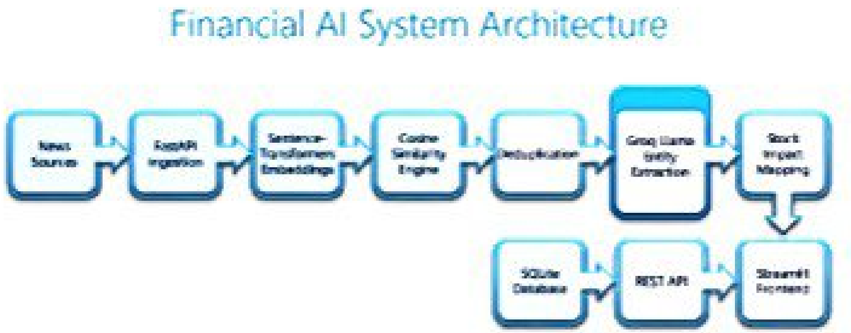
Analyzes sentiment and predicts price impact direction and magnitude.

Bullish/Bearish/Neutral · Magnitude: High/Med/Low

Architecture & Technology Stack

From ingestion to insights: end-to-end system flow and components

System Flow



Backend

- FastAPI, Uvicorn
- REST API, Swagger/ReDoc
- Pydantic models

AI / LLM

- Groq Llama (3.3-70B versatile)
- Entity extraction & reasoning
- Sentiment & impact

Embeddings & Similarity

- Sentence-Transformers MiniLM
- scikit-learn cosine similarity
- Duplicate threshold ≥ 0.85

Data & Storage

- SQLite (articles, entities, impacts)
- Embeddings as BLOB
- Pandas, NumPy

Frontend

- Streamlit UI
- Real-time insights dashboard
- Search & query tools

Pipelines

- Ingestion → Dedup → Extraction
- Stock impact mapping
- Context-aware query engine

Data model: **articles**, **entities**, **stock_impacts**, **deduplication_records**

Deep Dive: Key Pipelines

Step-by-step processes and confidence scores

Deduplication

1. Ingest article text
2. Embed with MiniLM (Sentence-Transformers)
3. Cosine similarity vs corpus
4. Threshold ≥ 0.85 → mark duplicate/canonical

Accuracy ≥ 95% · Avg dup sim > 0.85

Entity Extraction

1. LLM parse (Groq Llama)
2. Classify: companies, sectors, regulators, people, events
3. Assign confidence per entity
4. Persist entities in DB

Precision ≥ 90% · Companies 100% · Sectors 60–80%

Stock Impact Mapping

1. Link entities to tickers
2. Determine impact type (direct/sector/regulatory)
3. Assign confidence score
4. Attach sentiment & magnitude

Direct 100% · Sector 60–80% · Regulatory variable

Context-Aware Query Engine

1. NL input → semantic retrieval
2. Context expansion (company/sector/regulatory)
3. Synthesize response
4. Return articles & rationale

Response < 5s · Semantic ≥ 0.70

Performance Metrics & Results

Measured outcomes from real-time financial news processing

Deduplication Accuracy

95% Achieved

Duplicate threshold ≥ 0.85

Entity Extraction Precision

90% Achieved

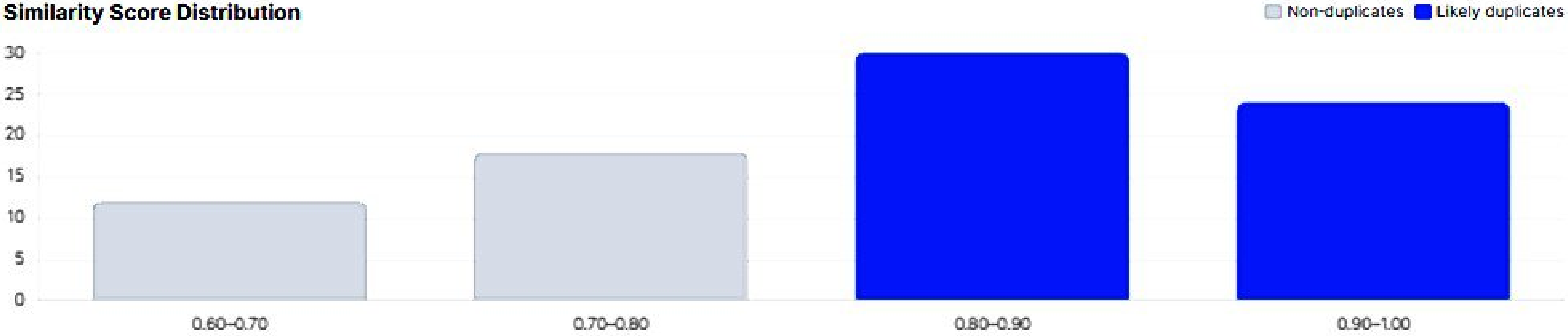
Companies 100%, Sectors 60–80%

Query Response Time

<5s Target met

Semantic search score ≥ 0.70

Similarity Score Distribution



Note: Duplicate detection uses cosine similarity with threshold **0.85**.

Use Cases & Applications

Five scenarios with example queries and user benefits

Trader Intelligence

For traders

Spot market-moving news and affected tickers fast.

🔍 "HDFC Bank news"

Risk Management

For risk teams

Track regulatory actions and sector-wide exposure.

🔍 "RBI policy changes"

Investment Research

For analysts

Analyze sentiment and price-impact patterns.

🔍 "Interest rate impact"

News Aggregation

For editors

Eliminate duplicates; consolidate unique stories.

🔍 "Banking sector update"

Market Monitoring

For PMs & desks

Track entity mentions and stock impacts in real time.

🔍 "IT services outlook"

Example Queries

- "HDFC Bank news" • "Banking sector update"
- "RBI policy changes" • "Interest rate impact"

Tip: Queries expand by company, sector, or regulator context.

Demo, API & Getting Started

Quick setup, key endpoints, docs, and next steps

≻_ Run locally

Backend (FastAPI)

```
cd backend
python3.11 -m venv venv && source venv/bin/activate
pip install -r requirements.txt
python app.py   # API at http://localhost:8000
```

Frontend (Streamlit)

```
pip install streamlit
streamlit run streamlit_app.py   # http://localhost:8501
```

Key API endpoints

- POST/GET `/articles`
- POST `/search`
- POST `/query`
- GET `/statistics`
- GET `/deduplication/stats`
- GET `/entities` , `/stock-impacts`

Documentation

Swagger /docs ReDoc /redoc Repository: {add link here}

→ Next steps: deployment & scaling

- Cloud deploy (API & Streamlit)
- Tune thresholds (0.85/0.70)
- Scale sources & storage
- Monitoring & alerting