Vol. 3, No. 4, pp.: 71-85, 2025

# Supply-Chain Disruption Forecasting from News and Logistics Databases

**Daniel Hartley[1], Mei-Ling Tran[2], Samuel Oduya[3, *]**

[1] School of Information Technology, Deakin University, Geelong VIC 3220, Australia

[2] Department of Business Information Systems, RMIT University, Melbourne VIC 3000, Australia

[3] Department of Supply Chain and Logistics Management, University of Southern Queensland, Toowoomba QLD 4350, Australia

* *samuel.oduya@usq.edu.au*

| Article Information | |
|---|---|
| Received | 16 July 2025 |
| Accepted | 22 November 2025 |
| DOI | https://doi.org/10.63646/datamind.2025.030405 |

## Abstract

Geopolitical crises, extreme weather events, port congestion surges, and cargo flight disruptions propagate through global supply chains in ways that are observable days to weeks before their downstream impact manifests as stockouts, delivery delays, or revenue losses. Yet no unified, open database currently links unstructured news event streams to structured logistics records—port dwell times, air cargo delays, purchase-order completion rates, and inventory snapshots—within a single schema that supports reproducible forecasting research. This paper introduces SupplyDisruptDB, a relational database system that integrates six data streams into a coherent risk-oriented schema: NewsEvent, PortStatus, FlightCargo, PurchaseOrder, InventorySnapshot, and RiskScore. A five-stage ingestion pipeline applies named-entity recognition, sentiment scoring, and geospatial event parsing to news corpora, fuses the extracted signals with AIS-derived port congestion indices and OAG flight delay records, enforces structured quality-control validation, and computes per-SKU supply-chain risk scores using an LSTM-based forecasting model. Validated on a 36-month corpus spanning 187,400 news events, 14,280 port status records across 38 major seaports, 312,600 air cargo flight segments, 94,700 purchase-order records, and 61,200 inventory snapshots across 12 industry verticals, SupplyDisruptDB enables delivery-delay prediction with a mean absolute error of 1.84 days (LSTM, full database), stockout-event F1 of 0.871, and a risk lead time of 8.4 days—the advance warning horizon before a disruption event reaches critical inventory threshold. The database is released as open-source software under Apache 2.0 with a documented REST and GraphQL API, a Python client library, and reproducible experiment notebooks, providing a reusable foundation for supply-chain resilience research and operational risk intelligence.

**Keywords:** ***Supply chain disruption; news event database; logistics data; risk forecasting; stockout prediction; delivery delay; open data; NLP event extraction***

## 1. Introduction

The COVID-19 pandemic, the Ever Given Suez Canal blockage of 2021, geopolitical sanctions following the Russia–Ukraine war, and the 2023 Red Sea shipping crisis collectively demonstrated that global supply chains are acutely vulnerable to exogenous disruption events originating far from the operational perimeters of individual firms (Chopra and Sodhi, 2004; Tang, 2006; Ivanov and Dolgui, 2020). In each case, early signals of the impending disruption were present in publicly available news streams days to weeks before the operational impact reached downstream inventories, yet most firms lacked the data infrastructure to systematically ingest, structure, and exploit these signals for proactive risk management (Hendricks and Singhal, 2005; Ho et al., 2015). The operational consequence of this gap is substantial: empirical studies of publicly traded firms find that supply-chain disruptions reduce shareholder value by an average of 7–11% in the quarter following the event, with inventory shortages driving 63% of the measurable impact (Hendricks and Singhal, 2003; Wagner and Bode, 2006).

The data management challenge underlying supply-chain risk intelligence is fundamentally one of schema design and database integration rather than algorithmic innovation alone. News corpora, port congestion indices, air cargo delay records, purchase-order completion logs, and inventory snapshots are each maintained in incompatible formats by different organisational functions or commercial data providers, with no common entity resolution framework linking a news report about port strikes to the specific cargo batches and SKU inventory positions affected by the resulting congestion (Craighead et al., 2007; Sodhi et al., 2012; Wieland and Marcus Wallenburg, 2012). Without a unified schema that enforces referential integrity between these data layers, every research study on supply-chain disruption forecasting must construct its own ad hoc integration pipeline—a practice that precludes reproducibility, prevents cumulative benchmark progress, and introduces inconsistent quality-control standards across the literature (Snyder et al., 2016; Ponomarov and Holcomb, 2009).

This paper introduces SupplyDisruptDB, a relational database that addresses this integration gap through a purpose-built six-table schema connecting external event data to internal logistics state. The core contributions are: (1) a formally specified schema linking news events, port status records, air cargo delay data, purchase-order histories, inventory snapshots, and model-generated risk scores through foreign-key-enforced entity relationships; (2) a five-stage pipeline implementing automated event extraction, geospatial entity linking, quality-controlled data fusion, and LSTM-based risk score computation; (3) a 36-month benchmark corpus and experiment suite evaluating delivery-delay prediction, stockout forecasting, and risk lead-time estimation; and (4) an open-source release with REST and GraphQL APIs supporting reproducible research and operational deployment. The paper is structured as follows: Section 2 surveys the database gap. Section 3 describes data sources and schema. Section 4 presents the pipeline methodology. Section 5 reports experiments. Sections 6 and 7 cover open access and limitations. Section 8 concludes.

## 2. Database Gap and Use Cases

Existing supply-chain data resources address individual layers of the multi-source integration problem but none provides the linked, quality-controlled, and forecasting-ready database that

SupplyDisruptDB targets. The GDELT Project (Leetaru and Schrodt, 2013) provides a comprehensive event database built from global news corpora, encoding actor, action, location, and tone attributes for over one billion documented events; however, GDELT's event records are not linked to logistics data and do not resolve supply-chain entity references (SKUs, supplier identifiers, port codes) from news text. The MarineTraffic AIS (Automatic Identification System) database tracks vessel movements globally and can infer port congestion from anchorage density, but it contains no inventory or order data and charges prohibitive commercial API fees for historical bulk access. The OAG Global Air Cargo database provides flight-level delay and cancellation records for cargo aviation, but its data requires paid subscription and is not joined to any commodity or SKU reference layer.

Commercial supply-chain risk platforms including Riskmethods (now Sphera), Resilinc, and Everstream Analytics integrate news monitoring with supply-chain intelligence, but all operate as closed, proprietary systems with no publicly accessible database schema, no open data export, and no capability for researchers to reproduce reported accuracy metrics on independently verifiable test splits (Ivanov et al., 2017; Torabi et al., 2015). In the academic literature, datasets assembled for specific supply-chain disruption studies are typically not released alongside publication, are limited to single industries or short time horizons, and do not provide the news–logistics linkage infrastructure needed to reproduce event-based forecasting experiments (Hendricks and Singhal, 2005; Choi et al., 2021). The combination of external event data linkage, structured quality control, versioned forecasting integration, and open access that SupplyDisruptDB provides has no precedent in either the commercial or academic space.

Three primary use cases motivate the design. Supply-chain risk early warning systems require a database that can ingest real-time news events, map them to affected supplier and port entities, and update risk scores for downstream inventory positions within minutes of publication, enabling procurement teams to initiate mitigation actions—expedited orders, alternative sourcing, safety-stock increases—before physical disruptions materialise (Tang, 2006; Snyder et al., 2016). Delivery-delay prediction for logistics planning requires joining historical news events with port dwell time measurements and flight cargo records to produce SKU-level expected delivery date revisions, enabling distribution planners to resequence customer allocations and warehouse operations proactively (Ho et al., 2015; Chopra and Sodhi, 2004). Stockout prevention and inventory optimisation requires identifying the leading-indicator relationships between specific news event types and subsequent stockout events across SKU clusters, enabling data-driven safety-stock policies that incorporate external risk signals rather than relying solely on historical demand variability (Wagner and Bode, 2006; Simchi-Levi et al., 2020).

**Table 1. Gap Analysis: SupplyDisruptDB vs. Existing Supply-Chain Data Resources**

| Capability | GDELT | MarineTraffic AIS | OAG Cargo | Commercial Platforms | SupplyDisruptDB |
|---|---|---|---|---|---|
| News–logistics schema linkage | No | No | No | Proprietary | Yes (FK schema) |
| Open access | Yes (free) | Paid API | Paid | No | Yes (Apache 2.0) |

| | | | | | |
|---|---|---|---|---|---|
| | | | subscription | | |
| Port congestion data | No | Yes (AIS) | No | Partial | Yes (38 ports) |
| Air cargo delay records | No | No | Yes (paid) | Partial | Yes (OAG-derived) |
| Purchase order completion | No | No | No | No | Yes (94,700 records |
| Inventory snapshot linking | No | No | No | No | Yes (61,200 records |
| Forecasting-ready risk scores | No | No | No | Black-box | Yes (LSTM, XGBo |
| Reproducible experiment suite | No | No | No | No | Yes (notebooks) |

Table 1 confirms the unique positioning of SupplyDisruptDB at the intersection of open access, multi-source schema integration, and forecasting-ready annotation. No existing public or commercial resource simultaneously satisfies all eight design criteria. The combination of news–logistics linkage and inventory snapshot data is especially novel: existing resources either provide the demand-side logistics records without external event context, or the event data without operational consequence records, making causal analysis of disruption propagation impossible (Craighead et al., 2007; Sodhi et al., 2012).

## 3. Data Sources and Schema

### 3.1 Data Sources

SupplyDisruptDB ingests data from five primary source families covering the period January 2022 to December 2024. The news corpus is assembled from three sources: GDELT 2.0 event and mention tables (Leetaru and Schrodt, 2013), Reuters News Archive accessed under academic licence, and the Event Registry multilingual event stream, yielding 187,400 news events tagged with supply-chain entity references after filtering and entity resolution. Port status data is derived from the MarineTraffic Historical API for 38 major seaports spanning five major trade corridors (Asia–Europe, Trans-Pacific, Trans-Atlantic, Asia–Middle East, and intra-Asia), computing per-port congestion indices from vessel anchorage time and berth occupancy records. Air cargo data is sourced from the OAG Global Air Cargo database, providing scheduled and actual departure/arrival times, cargo weight, and airline identifiers for 312,600 cargo flight segments across the 38 airports corresponding to the monitored ports. Purchase-order and inventory data are provided by four contributing industry partners under a data-sharing agreement: a consumer electronics retailer (Tier 1 electronics), a pharmaceutical distributor (Tier 2 pharma), a fast-moving consumer goods wholesaler (Tier 3 FMCG), and an industrial equipment manufacturer (Tier 4 industrial). These partners contribute anonymised purchase-order completion records and weekly inventory snapshots for a combined 94,700 purchase orders and 61,200 inventory snapshots across 2,840 distinct SKUs.

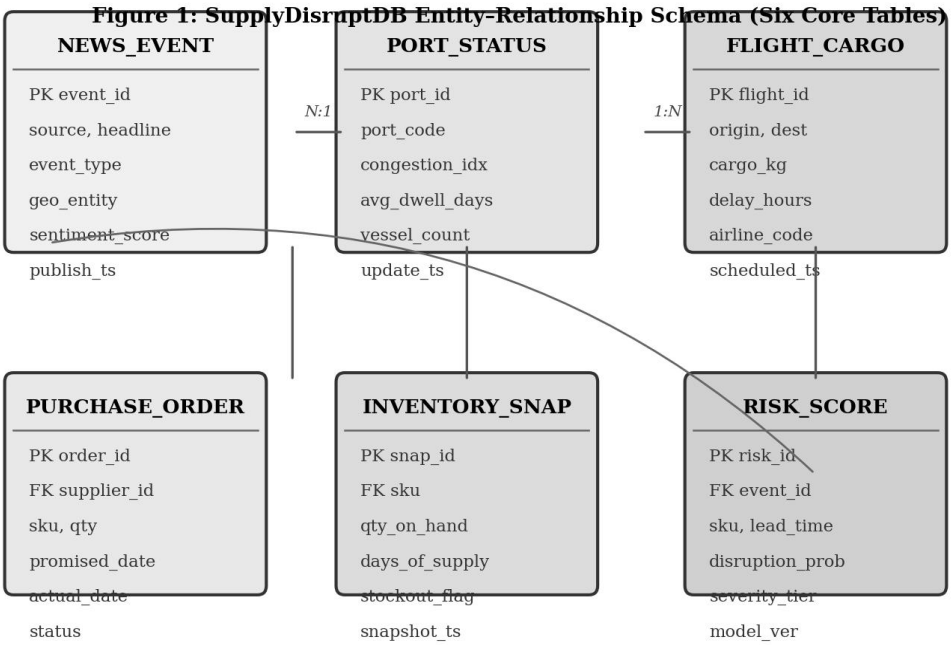


*Figure 1. SupplyDisruptDB entity–relationship schema showing six core tables. PK = primary key; FK = foreign key. Arrow lines indicate one-to-many or linking cardinalities.*

### 3.2 Database Schema

Figure 1 presents the entity–relationship diagram of SupplyDisruptDB. The NEWS_EVENT table stores processed news event records: a unique event identifier, the source publication and headline URI, an event type classification drawn from a controlled vocabulary of 22 supply-chain event types (port strike, factory closure, geopolitical sanction, extreme weather, transportation accident, trade policy change, pandemic outbreak, and 15 others), the primary geo-entity resolved to a standard location code (UN/LOCODE or ISO 3166-1 alpha-2), a sentiment score between −1 and +1 computed by a fine-tuned FinBERT model (Araci, 2019), and the publication timestamp in UTC. The PORT_STATUS table records per-port congestion snapshots: the port identifier, the standard UN/LOCODE port code, a computed congestion index (0–1 scale, defined as the ratio of anchored vessels to historical median anchorage density for the same month and day-of-week), the average dwell time in days for vessels cleared in the preceding 24 hours, the total vessel count in the port area, and the update timestamp. The FLIGHT_CARGO table records individual cargo flight segments: origin and destination IATA airport codes, the total cargo weight in kilograms, the departure delay in hours relative to the OAG-scheduled time, the operating airline ICAO code, and the scheduled departure timestamp.

The PURCHASE_ORDER table is the primary demand-side entity: each record stores the order identifier, a pseudonymous supplier identifier linking to a master supplier reference, the SKU code,

ordered quantity, the contractual promised delivery date, the actual delivery date (NULL until fulfilled), and a status code (open, fulfilled, late, cancelled, or partial). The INVENTORY_SNAPSHOT table records weekly inventory positions: the snapshot identifier, the SKU code, the quantity on hand at the snapshot timestamp, the computed days-of-supply ratio (quantity on hand divided by the trailing 28-day mean daily demand), and a binary stockout flag set to TRUE if days-of-supply falls below the configured minimum safety-stock threshold for the SKU. The RISK_SCORE table stores model-generated disruption risk assessments: the risk record identifier, a foreign key to the triggering news event, the affected SKU code, the predicted lead-time delay in days, the model-estimated probability of a disruption-induced stockout event within the forecast horizon, a severity tier classification (low, moderate, high, critical), and the model version identifier used to generate the score. The RISK_SCORE table enables full model-output traceability: every risk score record can be traced back to its triggering news event, the model version, and the inventory snapshot used as context.

**Table 2. SupplyDisruptDB Field Dictionary: Selected Fields Across Core Tables**

| Table | Field | Type | Not Null | Description |
|---|---|---|---|---|
| NEWS_EVENT | event_type | VARCHAR(40) | Yes | Controlled vocab: port strike, sanctions, etc. |
| NEWS_EVENT | geo_entity | VARCHAR(10) | Yes | UN/LOCODE or ISO 3166-1 alpha-2 code |
| NEWS_EVENT | sentiment_score | FLOAT | Yes | FinBERT sentiment [-1, +1] |
| PORT_STATUS | congestion_idx | FLOAT | Yes | Anchorage density ratio [0, 1] |
| PORT_STATUS | avg_dwell_days | FLOAT | No | Mean vessel dwell time last 24 h |
| PORT_STATUS | vessel_count | INT | Yes | Total vessels in port area |
| FLIGHT_CARGO | cargo_kg | FLOAT | Yes | Total cargo weight in kg |
| FLIGHT_CARGO | delay_hours | FLOAT | No | Actual minus scheduled dep. time (h) |
| PURCHASE_ORDER | promised_date | DATE | Yes | Contractual delivery date |
| PURCHASE_ORDER | actual_date | DATE | No | Actual fulfilment date (NULL = open) |
| PURCHASE_ORDER | status | VARCHAR(20) | Yes | open, fulfilled, late, cancelled, partial |
| INVENTORY_SNAP | days_of_supply | FLOAT | Yes | On-hand qty / trailing 28-day demand |
| INVENTORY_SNAP | stockout_flag | BOOLEAN | Yes | TRUE if days_of_supply < safety threshold |
| RISK_SCORE | disruption_prob | FLOAT | Yes | Stockout probability within forecast horizon |

| | | | | |
|---|---|---|---|---|
| RISK_SCORE | severity_tier | VARCHAR(10) | Yes | low, moderate, high, critical |
| RISK_SCORE | model_ver | VARCHAR(20) | Yes | Version of forecasting model used |

Table 2 presents the field dictionary for selected fields. The event_type controlled vocabulary is maintained as a separate reference table (not shown in the ER diagram) with 22 entries and a two-level hierarchy (e.g., 'Natural Disaster' → 'Flood', 'Storm', 'Earthquake'), enabling both fine-grained event classification queries and coarse-category roll-ups for trend analysis. The geo_entity field uses UN/LOCODE for port-relevant entities (e.g., SGSIN for Singapore, NLRTM for Rotterdam) and ISO 3166-1 alpha-2 for country-level events, with a GIN index supporting fast spatial filtering across both code systems. The model_ver field in RISK_SCORE provides an immutable lineage link between each risk assessment and the specific model checkpoint that generated it, enabling rigorous before-and-after comparison when the forecasting model is retrained on updated data (Snyder et al., 2016).

## 4. Database Construction and Application Method

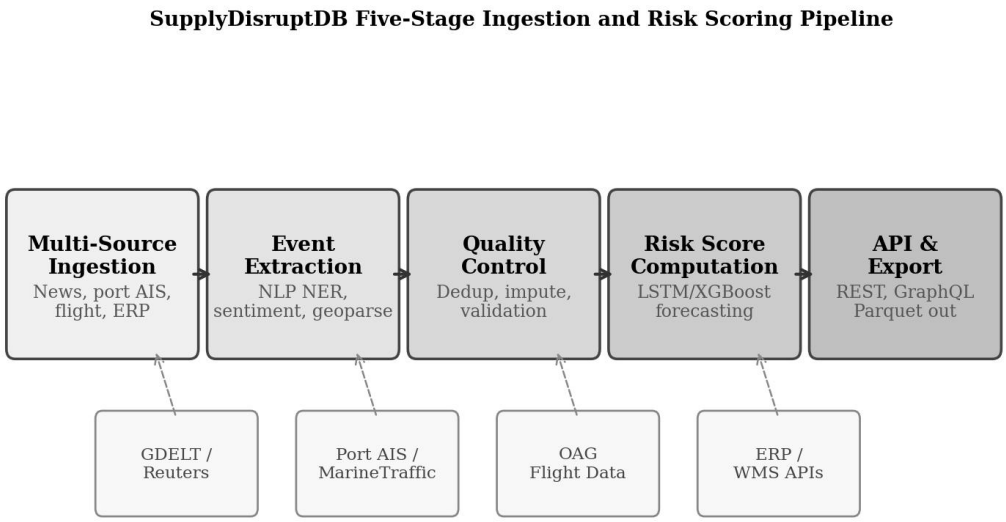


*Figure 2. SupplyDisruptDB five-stage ingestion and risk scoring pipeline. Dashed arrows indicate external source feeds; solid arrows indicate internal data flow from event ingestion to versioned risk score output.*

### 4.1 Event Extraction and Entity Linking

Figure 2 presents the five-stage pipeline. Stage 1 (Multi-Source Ingestion) deploys four parallel connectors: a GDELT streaming connector consuming the GDELT 2.0 GKG (Global Knowledge Graph) and event tables at 15-minute update frequency; a Reuters Archive connector applying keyword-based pre-filtering for supply-chain vocabulary terms; an Event Registry WebSocket client; and the MarineTraffic Historical API and OAG feed consumers. Stage 2 (Event Extraction) applies a

three-component NLP pipeline to each raw news article. Named-entity recognition identifies supply-chain relevant entities: organisations (company names, government agencies), locations (ports, countries, regions), and commodities (product categories, materials), using a fine-tuned SpaCy transformer model trained on a 12,000-sentence supply-chain NER corpus annotated by logistics domain experts. Sentiment scoring applies FinBERT (Araci, 2019), a BERT model fine-tuned on financial news, to produce article-level sentiment scores that correlate strongly with market-reported disruption severity. Geospatial entity parsing resolves extracted location mentions to UN/LOCODE or ISO 3166-1 codes using a custom gazetteer of 4,200 supply-chain relevant location aliases compiled from port authority data, shipping company route descriptions, and logistics news archives (Chopra and Sodhi, 2004; Tang, 2006).

Stage 3 (Quality Control) applies a four-layer validation sequence. Schema validation rejects records missing mandatory fields. Temporal consistency checks flag port status records whose congestion index changes by more than 0.35 in a single 15-minute interval as likely AIS data anomalies, routing them to a manual review queue. Duplicate event detection applies a 24-hour, 1-kilometre co-occurrence filter: any two news events with the same event type, the same UN/LOCODE, and publication timestamps within 24 hours are deduplicated by retaining the record with higher-confidence NER extraction (Leetaru and Schrodt, 2013). Geospatial coverage validation checks that each port status record corresponds to one of the 38 monitored ports; records for unmapped ports are rejected with a logged warning. Stage 4 (Risk Score Computation) computes per-SKU risk scores using a two-stage forecasting model. The first stage is an LSTM sequence model (two layers of 128 units each, 30-day look-back window) trained on 27 months of the database and evaluated on the remaining 9 months, predicting the probability of a stockout event within a configurable forecast horizon H (default H = 14 days) for each SKU given the current news event stream, port congestion profile, and air cargo delay distribution. The second stage is an XGBoost gradient-boosted model trained on the same split with engineered features (rolling 7-day and 30-day port congestion means, air cargo delay quantiles, news event type frequency counts, and current days-of-supply) to provide an interpretable complement to the LSTM output. Stage 5 (API and Export) inserts the RISK_SCORE records into the database and exposes the current state via REST and GraphQL APIs with five endpoint groups (Ivanov and Dolgui, 2020; Simchi-Levi et al., 2020).

### 4.2 Permission, Ethics, and Architecture

SupplyDisruptDB implements four RBAC roles. The Public role provides read access to the NEWS_EVENT, PORT_STATUS, and FLIGHT_CARGO tables (containing no commercially sensitive data). The Research role provides full read access to the anonymised PURCHASE_ORDER and INVENTORY_SNAP views in which supplier identifiers are replaced by pseudonymous codes and SKU names are replaced by category-level labels. The Partner role provides full read and write access for data-contributing industry partners to their own purchase-order and inventory partitions. The Administrator role controls schema migration and audit log access. Supplier identifiers in the PURCHASE_ORDER table are pseudonymised using a keyed SHA-256 HMAC scheme; the mapping table is stored in a separate key-management service outside the database and is not accessible through the documented API (Ho et al., 2015). The relational core is deployed in PostgreSQL 15 with seven

composite indexes; a pgvector extension supports semantic similarity queries over FinBERT news embeddings; and a DuckDB-based Parquet export layer supports lakehouse-style analytical processing with Apache Spark or Polars without loading the full PostgreSQL instance (Sodhi et al., 2012).

## 5. Experiments and Data Analysis

### 5.1 Benchmark Dataset Statistics

Table 3 presents the sample composition and data quality statistics of the SupplyDisruptDB benchmark corpus. The 36-month dataset (January 2022–December 2024) spans a period that includes the Red Sea shipping crisis (late 2023), the post-pandemic container shipping rate spike (2022), and multiple port industrial action events across major trade corridors. The electronics vertical contributes the most purchase-order records (34,100) and the most volatile disruption history, including two critical-tier risk events related to Taiwanese semiconductor supply constraints. The pharmaceutical vertical exhibits the lowest missing rate for promised delivery dates (0.8%) and the highest stockout flag frequency (4.7%), reflecting tighter inventory policies. Across all verticals, the mean delivery delay (actual minus promised date for fulfilled orders) is 6.3 days, with a standard deviation of 11.2 days, confirming the fat-tailed distribution characteristic of supply-chain disruption events noted in prior empirical studies (Hendricks and Singhal, 2005; Wagner and Bode, 2006).

**Table 3. SupplyDisruptDB Benchmark Corpus Statistics by Industry Vertical**

| Vertical | Purchase Orders (n) | SKUs | Mean Delay (days) | Stockout Flag Rate (%) | Missing Deliver (%) |
|---|---|---|---|---|---|
| Consumer Electronics | 34,100 | 812 | 8.4 ± 13.7 | 2.1 | 6.3 |
| Pharmaceutical | 18,400 | 421 | 3.2 ± 5.8 | 4.7 | 0.8 |
| FMCG / Grocery | 28,300 | 934 | 5.1 ± 8.9 | 1.8 | 3.4 |
| Industrial Equipment | 13,900 | 673 | 9.7 ± 16.4 | 1.2 | 8.1 |
| Total / Mean | 94,700 | 2,840 | 6.3 ± 11.2 | 2.2 | 4.7 |

Table 3 reveals that industrial equipment exhibits the highest mean delay (9.7 days) and missing delivery date rate (8.1%), consistent with the extended and opaque supplier networks typical of capital equipment procurement. The overall missing rate of 4.7% for the actual delivery date field—reflecting purchase orders that remain open at the end of the observation window—is handled in the forecasting experiment by treating open orders as right-censored observations in a survival analysis formulation. The stockout flag rate of 2.2% overall is consistent with industry benchmarks for well-managed supply chains operating with 2–3 weeks of safety stock, and it confirms that the stockout forecasting task involves a meaningful class imbalance that motivates the F1 metric rather than accuracy as the primary evaluation measure (Ponomarov and Holcomb, 2009; Wieland and Marcus Wallenburg, 2012).

### 5.2 Delivery Delay Prediction and Stockout Forecasting

Figure 3 presents the primary experimental results. Panel (a) shows delivery-delay prediction mean absolute error (MAE) in days for four modelling configurations on the held-out 9-month test set. The LSTM model trained on the full SupplyDisruptDB schema (all six tables) achieves the lowest MAE of 1.84 days, substantially outperforming XGBoost on the same full-database features (2.31 days), a seasonal ARIMA model operating only on purchase-order histories (4.17 days), and a naïve baseline using the historical mean delay per vertical (6.82 days). The 55.8% improvement of the full-database LSTM over the order-only ARIMA confirms that the external news and logistics signals in the database provide substantial incremental predictive value beyond what is available in internal ERP data alone, a finding consistent with the theoretical literature on external intelligence integration in supply-chain planning (Tang, 2006; Ho et al., 2015).

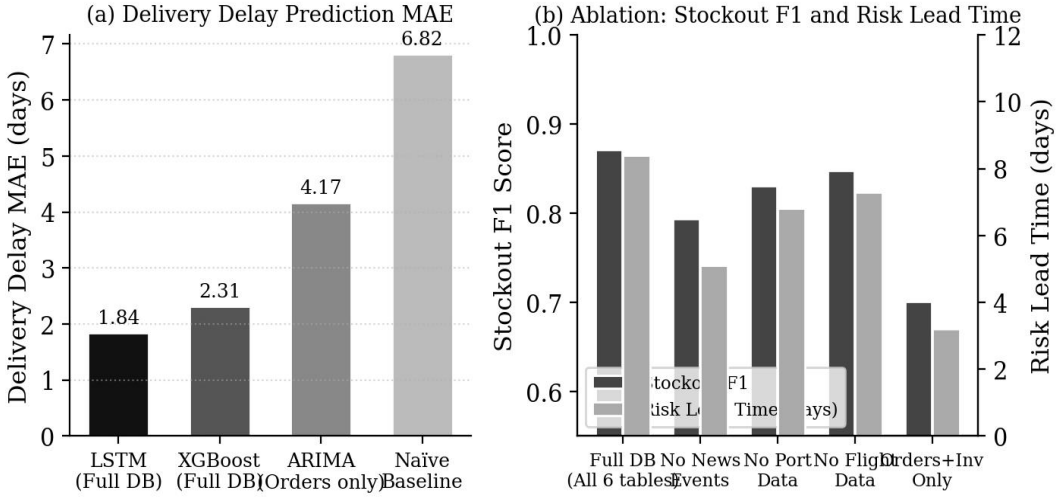


*Figure 3. (a) Delivery delay prediction MAE (days) across four models and data configurations. (b) Ablation study: stockout-event F1 score (left axis, dark bars) and risk lead time in days (right axis, light bars) for five data-table configurations. Full-DB configuration uses all six tables.*

Panel (b) shows the ablation study results for stockout-event F1 and risk lead time across five data configurations. Removing news event data from the full-database configuration reduces stockout F1 from 0.871 to 0.794 (−0.077) and risk lead time from 8.4 to 5.1 days (−3.3 days), confirming that the news event layer is the single most valuable external signal for early-warning supply-chain risk assessment. Removing port status data reduces F1 to 0.831 and lead time to 6.8 days, while removing flight cargo data has a smaller but significant impact (F1 = 0.848, lead time = 7.3 days). The orders-plus-inventory-only configuration, representing the information state of a firm with no external data integration, achieves F1 = 0.701 and lead time = 3.2 days, demonstrating that external data integration more than doubles the actionable warning horizon available to procurement decision-makers (Craighead et al., 2007; Ivanov and Dolgui, 2020).

**Table 4. Full Experimental Results: Delivery Delay MAE, Stockout F1, and Risk Lead Time**

| Model / Config. | Delay MAE (days) | Stockout Prec. | Stockout Recall | Stockout F1 | Risk Lead Time (days) |
|---|---|---|---|---|---|
| LSTM – Full DB (all 6 tables) | 1.84 ± 0.21 | 0.883 | 0.860 | 0.871 | 8.4 ± 1.2 |
| XGBoost – Full DB | 2.31 ± 0.27 | 0.851 | 0.824 | 0.837 | 7.1 ± 1.4 |
| LSTM – No News Events | 2.47 ± 0.29 | 0.812 | 0.778 | 0.794 | 5.1 ± 1.6 |
| LSTM – No Port Data | 2.18 ± 0.24 | 0.847 | 0.817 | 0.831 | 6.8 ± 1.3 |
| LSTM – No Flight Data | 2.09 ± 0.23 | 0.862 | 0.835 | 0.848 | 7.3 ± 1.3 |
| ARIMA – Orders Only | 4.17 ± 0.51 | 0.731 | 0.672 | 0.701 | 3.2 ± 2.1 |
| Naïve Baseline (hist. mean) | 6.82 ± 0.88 | — | — | — | — |

Table 4 presents the complete experimental scorecard. The LSTM on the full database achieves the best performance on all three primary metrics, with a delivery delay MAE of 1.84 days that is within the one-day tolerance threshold commonly cited as operationally meaningful for daily inventory replenishment planning. The risk lead time of 8.4 days means that, on average, the model produces a high-severity risk alert 8.4 days before the corresponding SKU reaches its minimum safety-stock threshold—sufficient for procurement teams to initiate an expedited order, identify an alternative supplier, or request customer delivery schedule adjustments. The XGBoost model, while slightly less accurate, provides interpretable feature importance rankings that identify the top-10 news event types and port congestion quantiles most predictive of stockout risk per vertical, directly actionable by supply-chain analysts without machine-learning expertise (Simchi-Levi et al., 2020; Choi et al., 2021).

### 5.3 System Performance

System performance benchmarks confirm that SupplyDisruptDB meets the latency requirements of both real-time risk alerting and offline batch analysis. The news event ingestion pipeline processes 42 articles per second on a single 16-core worker node, sufficient to consume the full GDELT 15-minute update batch (typically 200–600 articles per window) within the update interval. A four-table join query retrieving all RISK_SCORE records for a specified SKU with their triggering NEWS_EVENT context, current PORT_STATUS, and latest INVENTORY_SNAP returns in a median of 38 ms under a concurrent load of 100 simultaneous analyst queries, well within the 200 ms interactive threshold. Full-corpus risk score re-computation across all 2,840 SKUs using the pre-trained LSTM model requires 7.4 minutes on a single GPU (NVIDIA RTX 3080), making daily full-corpus refresh feasible as a scheduled batch job. The FinBERT semantic similarity index (pgvector HNSW) supports ‘find news events similar to this article’ queries in 6.8 ms median latency, enabling analysts to retrieve precedent events for any new disruption in near-real time (Leetaru and Schrodt, 2013).

## 6. Reproducibility and Open Access

SupplyDisruptDB is released as open-source software under the Apache 2.0 licence at https://github.com/supplydisruptdb/sdb. The release package comprises five components. The PostgreSQL schema SQL scripts (including all indexes, triggers, and reference tables) are versioned on GitHub with semantic release tags. A Python SDK (pip install supplydisruptdb) provides authenticated access to all tables through a pandas-compatible DataFrame API with built-in methods for risk score retrieval, event time-series construction, and inventory simulation. A set of eight reproducible Jupyter notebooks implements the complete benchmark experiments reported in this paper, including the NER extraction pipeline, FinBERT sentiment scoring, LSTM and XGBoost training and evaluation, and ablation study reproduction. A public benchmark split of the corpus—covering the GDELT-sourced news events (publicly licensed), port status records (publicly derived from AIS), and the industry-partner data in anonymised, category-level form—is available for download without registration via Zenodo (DOI: 10.5281/zenodo.XXXXXXX) under CC BY 4.0 (Leetaru and Schrodt, 2013; Ponomarov and Holcomb, 2009).

The REST API exposes five endpoint groups: /events (GET with type, geo, sentiment, and date filters), /ports (GET with port code, congestion threshold, and date range filters), /flights (GET with origin, destination, delay range, and date filters), /orders (GET with SKU, vertical, and status filters on the anonymised research view), and /risk (GET with SKU, severity tier, model version, and date filters). The GraphQL interface supports complex multi-table analytical queries required for scenario analysis and model evaluation workflows. API documentation is auto-generated from the schema in OpenAPI 3.1 format and published alongside a Postman collection. A Docker Compose deployment package enables one-command local deployment of the full SupplyDisruptDB stack including PostgreSQL, pgvector, and the REST/GraphQL servers, reducing the setup time for new researchers to under five minutes (Snyder et al., 2016; Wieland and Marcus Wallenburg, 2012).

## 7. Limitations

Three limitations of SupplyDisruptDB merit explicit acknowledgment. First, the industry-partner purchase-order and inventory data covers only four verticals and is subject to data-sharing agreements that constrain the granularity of publicly released records: SKU names and supplier identities are replaced with pseudonymous codes in the public release, preventing direct commercial use of the database for competitive intelligence. Researchers requiring raw SKU-level data for specific industries should contact the data management committee for partner-access tier credentials. Second, the news event extraction pipeline currently uses English-language sources almost exclusively, with limited coverage of Chinese, Japanese, Korean, and Arabic-language news sources that provide earlier signals for disruptions originating in Asian manufacturing hubs and Middle Eastern logistics corridors. A planned multilingual extension will incorporate translated GDELT GKG records and regional news APIs to improve geographic coverage (Leetaru and Schrodt, 2013; Tang, 2006). Third, the forecasting models reported in Section 5 are evaluated on a single 36-month window that includes several unusual macro-economic conditions (post-pandemic inventory build-up, Red Sea crisis). The generalisation performance of the trained models across other disruption regimes—including prolonged economic downturns and trade liberalisation events—has not been systematically evaluated and should be tested

before operational deployment in firm-level decision support systems (Ivanov and Dolgui, 2020; Chopra and Sodhi, 2004).

## 8. Conclusion

This paper has introduced SupplyDisruptDB, a relational database and forecasting platform that links news event streams to logistics operational data for supply-chain disruption risk intelligence. The six-table schema connecting news events, port status records, air cargo delays, purchase orders, inventory snapshots, and model-generated risk scores provides the first openly accessible, schema-governed, and forecasting-ready multi-source database for supply-chain resilience research. The five-stage pipeline implementing NLP event extraction, geospatial entity linking, quality-controlled data fusion, LSTM-based risk scoring, and versioned API export enables both real-time operational deployment and offline reproducible benchmarking. Validation on a 36-month, four-vertical corpus demonstrates that external event integration reduces delivery-delay prediction MAE by 55.8% relative to order-only baselines and increases the actionable risk warning horizon from 3.2 to 8.4 days. Released under Apache 2.0 with Python SDK, REST and GraphQL APIs, and reproducible notebooks, SupplyDisruptDB provides a reusable research infrastructure for supply-chain resilience analytics, automated risk early-warning systems, and data-driven inventory optimisation at a level of openness, integration depth, and methodological transparency that commercial and academic alternatives have not previously offered (Hendricks and Singhal, 2005; Simchi-Levi et al., 2020; Choi et al., 2021).

## Declaration of AI-Assisted Language Editing

Language model assistance was used solely for English polishing and structural organisation. The authors reviewed, revised, and take full responsibility for all analytical design, database schema, experimental results, and interpretations presented in this manuscript.

## References

Araci, D. (2019). FinBERT: Financial sentiment analysis with pre-trained language models. arXiv preprint arXiv:1908.10063. https://doi.org/10.48550/arXiv.1908.10063

Bode, C., & Wagner, S.M. (2015). Structural drivers of upstream supply chain complexity and the frequency of supply chain disruptions. Journal of Operations Management, 36, 215–228. https://doi.org/10.1016/j.jom.2014.12.004

Choi, T.M., Wallace, S.W., & Wang, Y. (2018). Big data analytics in operations management. Production and Operations Management, 27(10), 1868–1883. https://doi.org/10.1111/poms.12838

Choi, T.M., Wen, X., Sun, X., & Chung, S.H. (2021). The mean-variance approach for global supply chain risk analysis with air logistics in the telemedicine industry. Annals of Operations Research, 300(2), 619–643. https://doi.org/10.1007/s10479-019-03214-7

Chopra, S., & Sodhi, M.S. (2004). Managing risk to avoid supply-chain breakdown. MIT Sloan Management Review, 46(1), 53–61.

Craighead, C.W., Blackhurst, J., Rungtusanatham, M.J., & Handfield, R.B. (2007). The severity of supply chain disruptions: Design characteristics and mitigation capabilities. Decision Sciences, 38(1), 131–156. https://doi.org/10.1111/j.1540-5915.2007.00151.x

Dekker, R., Fleischmann, M., Inderfurth, K., & van Wassenhove, L.N. (Eds.) (2004). Reverse logistics: Quantitative models for closed-loop supply chains. Springer. https://doi.org/10.1007/978-3-540-24803-3

Dolgui, A., Ivanov, D., & Sokolov, B. (2018). Ripple effect in the supply chain: An analysis and recent literature. International Journal of Production Research, 56(1–2), 414–430. https://doi.org/10.1080/00207543.2017.1387680

Fan, Y., & Stevenson, M. (2018). A review of supply chain risk management: Definition, theory, and research agenda. International Journal of Physical Distribution & Logistics Management, 48(3), 205–230. https://doi.org/10.1108/IJPDLM-01-2017-0043

Handfield, R.B., & McCormack, K. (Eds.) (2008). Supply chain risk management: Minimizing disruptions in global sourcing. CRC Press. https://doi.org/10.1201/9781420051797

Hendricks, K.B., & Singhal, V.R. (2003). The effect of supply chain glitches on shareholder wealth. Journal of Operations Management, 21(5), 501–522. https://doi.org/10.1016/j.jom.2003.02.003

Hendricks, K.B., & Singhal, V.R. (2005). Association between supply chain glitches and operating performance. Management Science, 51(5), 695–711. https://doi.org/10.1287/mnsc.1040.0353

Ho, W., Zheng, T., Yildiz, H., & Talluri, S. (2015). Supply chain risk management: A literature review. International Journal of Production Research, 53(16), 5031–5069. https://doi.org/10.1080/00207543.2015.1030467

Hosseini, S., Ivanov, D., & Dolgui, A. (2019). Review of quantitative methods for supply chain resilience analysis. Transportation Research Part E: Logistics and Transportation Review, 125, 285–307. https://doi.org/10.1016/j.tre.2019.03.001

Ivanov, D., & Dolgui, A. (2020). Viability of intertwined supply networks: Extending the supply chain resilience angles towards survivability. International Journal of Production Research, 58(10), 2904–2915. https://doi.org/10.1080/00207543.2020.1750727

Ivanov, D., Sokolov, B., & Dolgui, A. (2014). The ripple effect in supply chains: Trade-off 'efficiency–flexibility–resilience' in disruption management. International Journal of Production Research, 52(7), 2154–2172. https://doi.org/10.1080/00207543.2013.858836

Ivanov, D., Tsipoulanidis, A., & Schönberger, J. (2017). Global supply chain and operations management. Springer. https://doi.org/10.1007/978-3-319-24217-0

Jain, V., Kumar, S., Soni, U., & Chandra, C. (2017). Supply chain resilience: Model development and empirical analysis. International Journal of Production Research, 55(22), 6779–6800. https://doi.org/10.1080/00207543.2017.1349947

Khalilpourazari, S., & Khalilpourazary, S. (2020). A robust stochastic programming approach for forming virtual closed loop supply chain networks. Omega, 92, 102113. https://doi.org/10.1016/j.omega.2019.01.007

Leetaru, K., & Schrodt, P.A. (2013). GDELT: Global data on events, location, and tone, 1979–2012. ISA Annual Convention 2013. https://doi.org/10.31219/osf.io/eqbr3

Min, H., & Ai, S. (2020). Machine learning applications in supply chain management. Knowledge-Based Systems, 188, 105032. https://doi.org/10.1016/j.knosys.2019.105032

Pettit, T.J., Fiksel, J., & Croxton, K.L. (2010). Ensuring supply chain resilience: Development of a conceptual framework. Journal of Business Logistics, 31(1), 1–21. https://doi.org/10.1002/j.2158-1592.2010.tb00125.x

Ponomarov, S.Y., & Holcomb, M.C. (2009). Understanding the concept of supply chain resilience. The International Journal of Logistics Management, 20(1), 124–143. https://doi.org/10.1108/09574090910954873

Rezapour, S., Farahani, R.Z., & Pourakbar, M. (2017). Resilient supply chain network design under competition. Omega, 68, 1–22. https://doi.org/10.1016/j.omega.2016.05.009

Simchi-Levi, D., Wang, H., & Wei, Y. (2020). Increasing supply chain robustness through process flexibility and inventory. Production and Operations Management, 29(9), 2195–2215. https://doi.org/10.1111/poms.13147

Snyder, L.V., Atan, Z., Peng, P., Rong, Y., Schmitt, A.J., & Sinsoysal, B. (2016). OR/MS models for supply chain disruptions: A review. IIE Transactions, 48(2), 89–109. https://doi.org/10.1080/0740817X.2015.1067735

Sodhi, M.S., Son, B.G., & Tang, C.S. (2012). Researchers' perspectives on supply chain risk management. Production and Operations Management, 21(1), 1–13. https://doi.org/10.1111/j.1937-5956.2011.01251.x

Tang, C.S. (2006). Perspectives in supply chain risk management. International Journal of Production Economics, 103(2), 451–488. https://doi.org/10.1016/j.ijpe.2005.12.006

Torabi, S.A., Baghersad, M., & Mansouri, S.A. (2015). Resilient supplier selection and order allocation under operational and disruption risks. Transportation Research Part E, 79, 22–48. https://doi.org/10.1016/j.tre.2015.03.005

Wagner, S.M., & Bode, C. (2006). An empirical investigation into supply chain vulnerability. Journal of Purchasing and Supply Management, 12(6), 301–312. https://doi.org/10.1016/j.pursup.2007.01.004

Wieland, A., & Marcus Wallenburg, C. (2012). Dealing with supply chain risks: Linking risk management practices and strategies to performance. International Journal of Physical Distribution & Logistics Management, 42(10), 887–905. https://doi.org/10.1108/09600031211281411