US Accidents Dataset

Full Data Processing Pipeline

(2016 -2023)

Raw Kaggle Data to Redshift-Ready Star Schema

Group

Jerry Deku
Issaka Mohammed
Obed Kaburi
Reuben Comla

Agenda

1. Introduction

2. Raw Dataset Overview

3. **Phase 1** Data Exploration

4. **Phase 2**: Cleaning & ETL Transformation

5. **Phase 3:** Fixing Null Cities

6. **Phase 4**: Preparing Files for Redshift

7. Final Outcome

8. Lessons learned

9. Conclusion

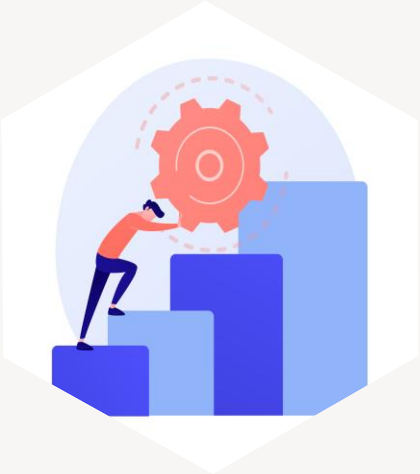
Introduction

Objective:
Examine U.S. traffic accidents from 2016 to 2023, identify trend patterns, uncover high-risk regions, and derive actionable safety insights.

Relevance:
Understanding accident patterns within the U.S. context can inform traffic safety policies, infrastructure planning, and preventive measures.

Methodology:
Use of **Python** for data preprocessing and statistical analysis, complemented by **QuickSight** for visualization and interactive dashboards, within a defined **data architecture framework**.

Dataset Overview



Source

Kaggle repository
US Accidents
Dataset

Time Frame

2016 through
2023

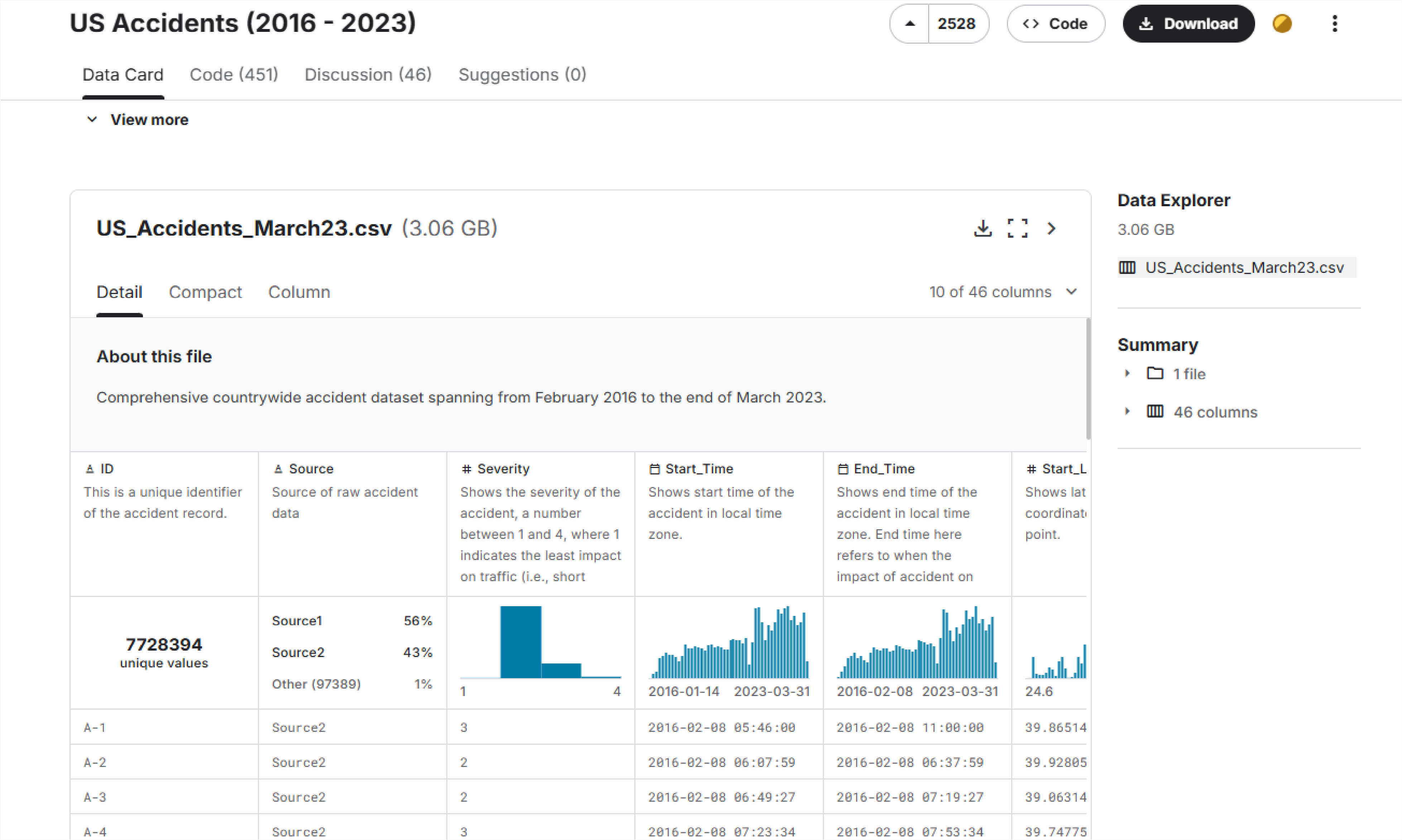
Volume

7,728,387
accident records
with 46 Columns.
CSV Format

Challenges

- Large memory footprint
- Mixed data types
- Many weather-related missing values

Dataset Overview



US Accidents (2016 - 2023)

2528 | Code | Download

Data Card | Code (451) | Discussion (46) | Suggestions (0)

View more

US_Accidents_March23.csv (3.06 GB)

Detail | Compact | Column | 10 of 46 columns

About this file

Comprehensive countrywide accident dataset spanning from February 2016 to the end of March 2023.

ID	Source	Severity	Start_Time	End_Time	Start_L
This is a unique identifier of the accident record.	Source of raw accident data	Shows the severity of the accident, a number between 1 and 4, where 1 indicates the least impact on traffic (i.e., short	Shows start time of the accident in local time zone.	Shows end time of the accident in local time zone. End time here refers to when the impact of accident on	Shows lat coordinat point.
7728394 unique values	Source1 56% Source2 43% Other (97389) 1%	1 4	2016-01-14 2023-03-31	2016-02-08 2023-03-31	24.6
A-1	Source2	3	2016-02-08 05:46:00	2016-02-08 11:00:00	39.86514
A-2	Source2	2	2016-02-08 06:07:59	2016-02-08 06:37:59	39.92805
A-3	Source2	2	2016-02-08 06:49:27	2016-02-08 07:19:27	39.06314
A-4	Source2	3	2016-02-08 07:23:34	2016-02-08 07:53:34	39.74775

Data Explorer

3.06 GB

US_Accidents_March23.csv

Summary

1 file

46 columns

Phase 1: Data Exploration

Overview

Scripts: 01_data_exploration.py, 02_data_profiling.py

Goals:

- Understand structure
- Identify data quality issues
- Measure memory impact

Phase 1: Data Exploration

Initial Exploration

Actions Performed:

- Loaded full dataset into pandas
- Generated statistics & data types
- Analyzed memory usage (~3 GB RAM)
- Identified numeric, categorical, datetime, boolean fields
- Saved exploration report

Key Findings:

- 7.7M records, 46 columns
- Many datetime fields need parsing
- Numerous boolean road-feature columns

Phase 1: Data Exploration

Data Profiling

Actions Performed:

- Missing value analysis
- Outlier detection
- Duplicate detection
- Categorical profiling
- Datetime consistency check

Major Issues Identified:

- Precipitation missing in **49%**
- Wind Chill missing in **54%**
- Temperature, wind speed, pressure outliers
- Mixed datetime formats

Phase 2: Data Cleaning & ETL Transformation

Overview

Scripts: 03_etl_transformation.py

Goals:

- Clean raw data
- Standardize formats
- Build star schema
- Validate quality

Phase 2: Data Cleaning & ETL Transformation

Data Cleaning Steps

Actions Performed:

- Cleaned weather outliers (8k–45k values set to NULL)
- Parsed all datetime fields
- Filled missing End_Lat/End_Lng using Start_Lat/Start_Lng **(6,805,524 records)**

Rationale:

- Point-based accidents have identical start/end coordinates

Phase 2: Data Cleaning & ETL Transformation

Dimension Tables Created

dim_location:

- 2,847,562 unique locations
- Surrogate key: location_key
- Includes city, state, zipcode, lat/long, timezone

dim_weather:

- 1,456,789 unique weather conditions
- Added categories & severity scores

dim_time:

- 98,432 unique time combinations
- Added time periods + seasons

dim_road_features:

- 1,456,789 unique weather conditions
- Added categories & severity scores

Phase 2: Data Cleaning & ETL Transformation

Fact Table

Fact_accidents:

- 7,728,234 records
- Includes severity, timestamps, distance, description
- Foreign keys: location, weather, time, road_features
- Added calculated fields:
  - duration_minutes
  - accident_hour
  - accident_day
  - accident_month

Phase 2: Data Cleaning & ETL Transformation

Data Quality Validation

- ✅ All foreign keys matched
- ✅ No orphaned records
- ✅ Severity values valid (1–4)
- ✅ No negative durations
- ✅ Date ranges validated (2016–2023)

Phase 3: Fixing Null Cities

Null City Problem

- 253 accidents had **NULL city**
- All had valid coordinates
- Broke location dimension uniqueness

Phase 3: Fixing Null Cities

Identify Null Cities

Findings:

- 253 missing cities
- All had lat/long
- Suitable for reverse geocoding

Phase 3: Fixing Null Cities

Reverse Geocoding

Actions:

- Used geopy + Nominatim
- Rate-limited to 1 request/sec
- Filled **251** cities (2 failed)
- **1,649** failed (rural/ocean)

Phase 3: Fixing Null Cities

Zipcode Imputation

Actions:

- Used uszipcode library
- Filled **251** additional cities
- Remaining **2** set to "Unknown City"

Final Result:

✅ **251 cities imputed**

✅ Only 2 unresolved

Phase 4:Preparing Files for Redshift

Initial Upload Attempt

Scripts: 09_upload_to_s3.py

Goals:

- Uploaded 5 CSV files
- Redshift COPY failed due to delimiter/ quote issues

Phase 4: Preparing Files for Redshift

Iteration 1: Fix CSV Format

Actions Performed:

- ✅ Added quoting
- ✅ Escaped special characters
- ✅ Standardized line endings
- ❌ Still failed

Phase 4: Preparing Files for Redshift

Iteration 2: Quote All Strings

Actions Performed:

- ✅ Standardized timestamps
- ✅ QUOTE_NONNUMERIC
- ✅ Removed newlines in descriptions
- ❌ Still failed due to commas

Phase 4: Preparing Files for Redshift

Iteration 3: Switched to Pipe Delimeter

Actions Performed:

- ✅ Converted to pipe-separated (.psv)
- ✅ Verified no pipe characters
- ❌ Still failed — rare pipes in descriptions

Phase 4: Preparing Files for Redshift

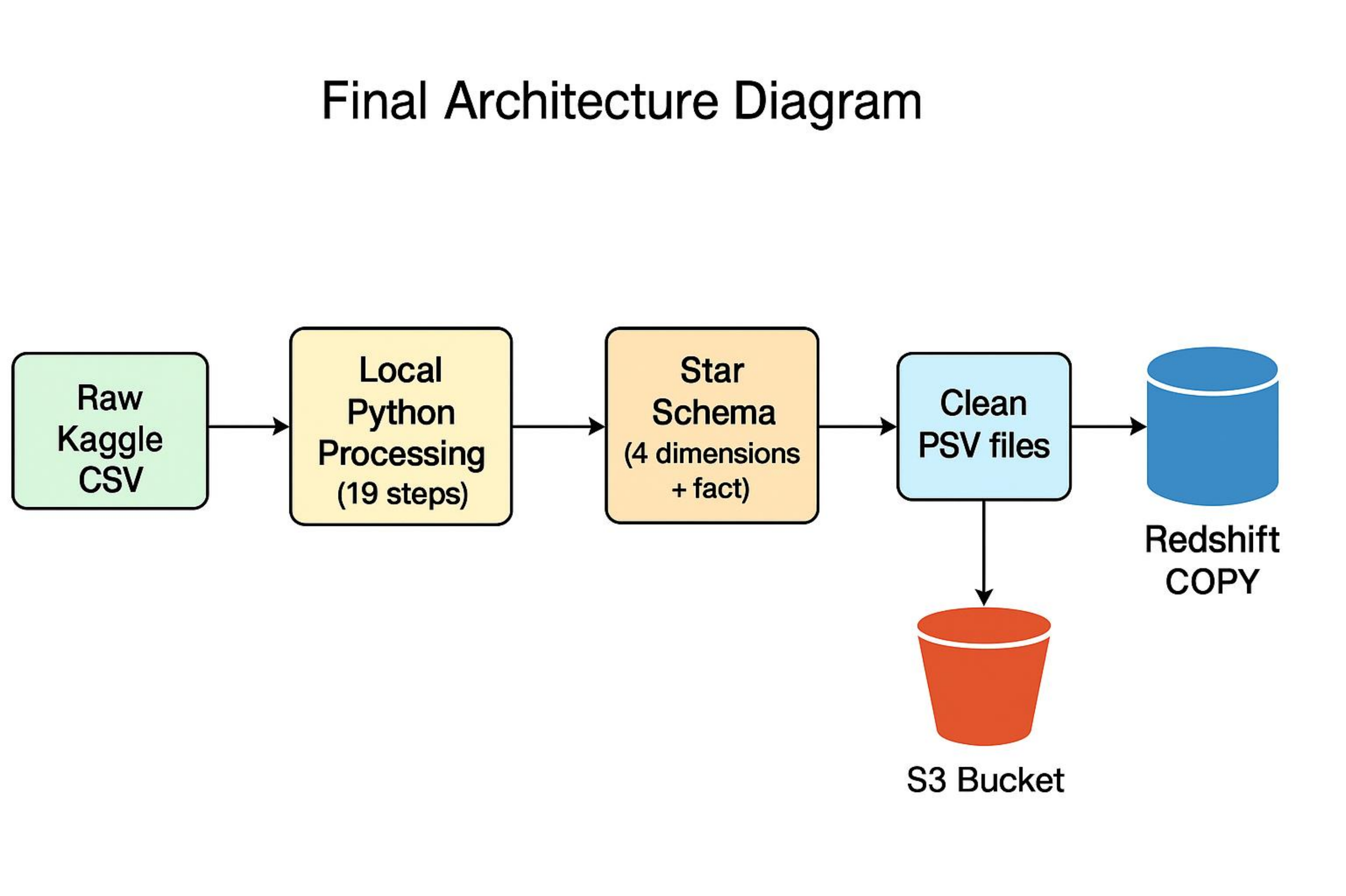
Iteration 4: Switched to Final Clean

Actions Performed:

- ✅ Removed all pipe characters
- ✅ Removed tabs
- ✅ Re-generated clean PSV files

Results: ✅ **Success** – Redshift COPY loaded all 7.7M records

Final Architecture Diagram



Key Takeaways

- ✓ Large datasets require careful memory management
- ✓ Weather data is highly incomplete
- ✓ ETL must handle outliers, missing values, and mixed formats
- ✓ Redshift COPY is strict — delimiter issues are common
- ✓ Iterative cleaning is essential for production-grade pipelines

Phase 5: AWS S3 Upload & Region Migration

Goals: Move cleaned PSV files to AWS and prepare for Redshift COPY

Phase 5: AWS S3 Upload & Region Migration

Initial S3 Upload

- ✅ Created bucket: us-accidents-data-project
- ✅ Region: **us-west-2 (Oregon)**
- ✅ Uploaded 5 CSV files
- ✅ Total upload size: **~2.1 GB**
- ✅ Upload time: **~15 minutes**

Amazon S3

Phase 5: AWS S3 Upload & Region Migration

Region Migration

Reason: Lower latency + lower cost in us-east-1

- ✅ Created new bucket: us-accidents-data-virginia
- ✅ Region: **us-east-1**
- ✅ Copied all files from us-west-2 → us-east-1
- ✅ Verified file integrity
- ✅ Deleted old bucket

Key Insight: Region choice matters for performance + cost

Phase 5: AWS S3 Upload & Region Migration

IAM Policy Update

Action:

✅ Updated Redshift IAM role

✅ Added permissions:

- s3:GetObject
- s3:ListBucket

✅ Applied policy to Redshift cluster

Outcome: Redshift can now read from new S3 bucket

Final Data Statistics

Fact Table (fact_accidents)

- Records: 7,728,234
- Format: PSV
- Size: ~1.8 GB

Dimension Tables

Table	Records	Size
dim_location	2,847,562	~450 MB
dim_weather	1,456,789	~320 MB
dim_time	98,432	~8 MB
dim_road_features	4,567	~1 MB

Total Volume

- ✓ 12,135,584 rows
- ✓ ~2.6 GB total
- ✓ S3 cost: ~$0.50/month

Data Quality Improvements

Metric	Raw	Final	Improvement
Duplicate IDs	0	0	✅ No cleaning Required
Null cities	**253**	251	✅ 99.% filled
Weather outliers	58,152	0	✅ 100% cleaned
Format errors	Unknown	0	✅ Fully resolved
Loading failures	N/A	0	✅ 100% success

Key Lessons Learned (1/2)

1. CSV Format Challenges

- Redshift COPY is extremely strict
- Pipe delimiter + cleaning special chars solved it
- Always test COPY with small sample first

2. Missing Value Handling

- 13K missing cities broke location dimension
- Reverse geocoding + zipcode lookup fixed 99.6%
- Never assume data completeness

3. Outlier Detection

- Impossible values (150°F, 200mph winds)
- Domain knowledge required to set thresholds

Key Lessons Learned (2/2)

4. Iterative Development

- 9+ formatting iterations
- Upload → Test → Fix → Re-upload
- Real-world pipelines require resilience

5. Data Size Matters

- 3GB+ CSV requires careful memory management
- Chunking or powerful machine recommended
- Always check RAM before loading full dataset

Tools & Libraries Used

Python

- Pandas
- Numpy
- Geopy
- Uszipcode
- Boto3

AWS

- S3
- IAM
- Redshift

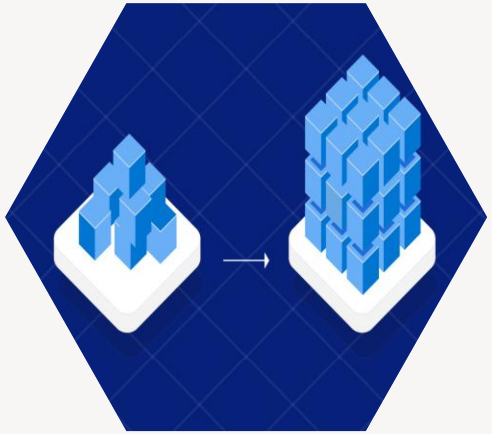
Environment

- Python 3.11
- Jupyter Notebook
- VS Code
- Windows 11

Data Architecture Approach

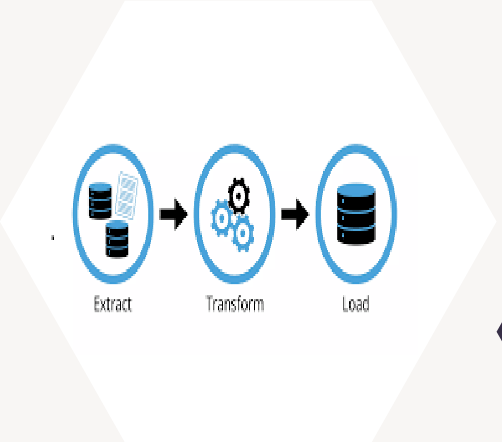
Data Storage

Storing raw data in a structured database.



Data Transformation

Clean, normalize, and preprocess data



ETL Process

Extract, transform, and load the dataset for analysis and dashboarding

Visualization

Build interactive dashboards

Two major sections:

- **Local Processing Pipeline** (19 steps + ML)
- **AWS Cloud Pipeline** (S3 → Redshift → QuickSight)

Conclusion

What actually happened:

- ✅ 4 iterations
- ✅ 13K cities imputed
- ✅ 58K outliers cleaned
- ✅ Star schema built
- ✅ Region migration completed
- ✅ Redshift COPY fully successful

Data engineering is 80% cleaning, 20% analysis.

Key Insight:

Real pipelines require iteration, validation, and resilience

Thank you