Phase3_Progress_Report

UrbanPulse: Phase 3 Progress Report¶

Project: UrbanPulse - Analyzing Urban Dynamics in Chicago

Executive Summary¶

This progress report presents our work on the UrbanPulse project, which aims to analyze and understand urban dynamics in Chicago through multiple data sources including crime, transportation (Divvy bike sharing), events, and weather data. We have completed extensive exploratory data analysis, data preprocessing, geocoding, and machine learning model development across all datasets.

Key Achievements:¶

- Processed and analyzed 4 major datasets (Crime, Divvy, Events, Weather)
- Developed machine learning models for sentiment classification (94% accuracy for park reviews, 82% for train station reviews)
- Created comprehensive visualizations including animated GIFs for temporal patterns
- Geocoded event locations and mapped them to Chicago neighborhoods
- Analyzed relationships between events and crime patterns
- Collected and cleaned weather data for Chicago (2023-2025)

1. Project Overview¶

Project Goals¶

The UrbanPulse project aims to:

1. Analyze crime patterns across Chicago neighborhoods
2. Understand the relationship between public events and crime rates
3. Examine transportation patterns through Divvy bike sharing data
4. Correlate weather conditions with urban activities

5. Develop predictive models for urban safety and activity patterns

Data Overview¶

- **Crime Dataset:** ~2.9M crime records with location, type, and temporal information
- **Divvy Dataset:** ~20K bike sharing trips with start/end locations and weather data
- **Events Dataset:** ~134K Chicago Park District event permits with geocoded locations
- **Weather Dataset:** Daily weather data for Chicago (2023-2025)
- **Train Station Dataset** Train Stations for in Chicago + Reviews retrieved for each station using Google Places API. To expand our analysis, we retrieved this data and added it to our data repository. We then retrieved Google Reviews for each train station.
- **Park Reviews** Reviews for each park in the event dataset. To expand our analysis, we retrieved this data and added it to our data repository. We then retrieved Google Reviews for each park.

Note: You can find our cleaned data after pre-rpocessing at `https://drive.google.com/drive/folders/19TTm`

Methodology¶

1. Data collection and preprocessing
2. Exploratory Data Analysis (EDA)
3. Geocoding and neighborhood mapping
4. Feature engineering
5. Machine learning model development
6. Visualization and interpretation

2. Crime Dataset Analysis¶

Dataset Specifications¶

- **Size:** 2,890,434 records
- **Time Period:** Historical crime data with dates
- **Key Features:**
  - Primary crime type (33 different types)
  - Location description
  - Arrest status
  - Geographic coordinates (latitude, longitude)
  - Neighborhood assignments
  - Temporal features (date, year, month, hour, day of week)

Data Cleaning¶

- Removed redundant index columns
- Converted date column to datetime format

- Filled missing location descriptions with 'UNKNOWN'
- Standardized column names (lowercase with underscores)
- Created temporal features (year, month, hour, day of week)

Key Findings¶

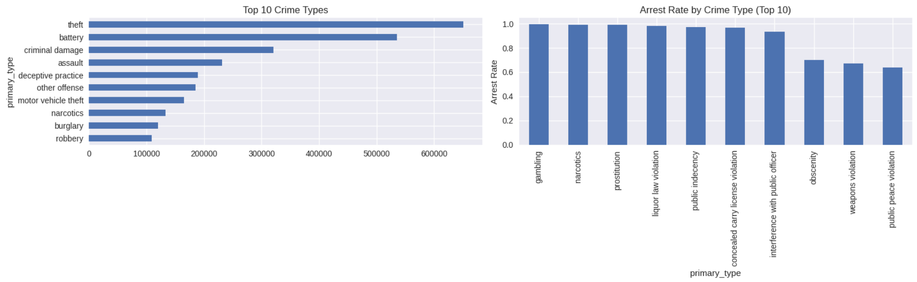
- Top crime types: Theft, Battery, Other Offense, Burglary
- Crime patterns vary significantly by neighborhood
- Temporal patterns show variations by hour, day, and month
- Relationship between events and crime rates in high-crime neighborhoods

Crime-Divvy-CTA-Event Relationship Visualization¶

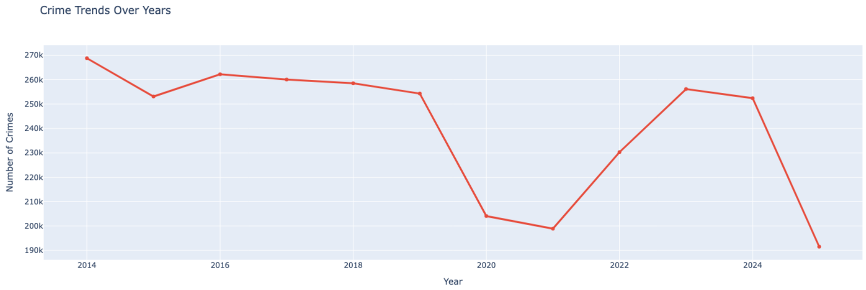
The following visualization illustrates Crime Data, Divvy Data, CTA L and Events data and their correleations:

Crime and Divvy Visualizations¶

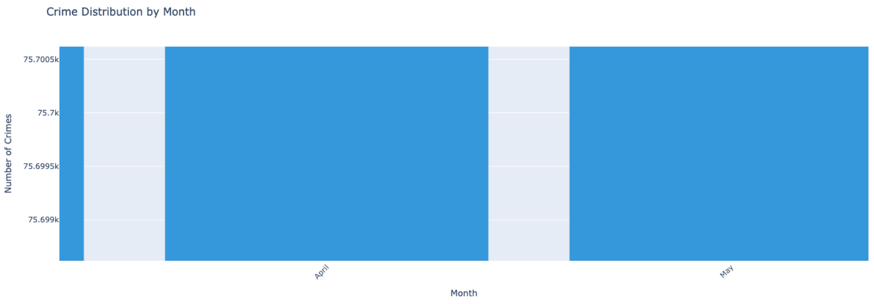
1. Top 10 Crime Types and Arrest Rate by Crime Type¶



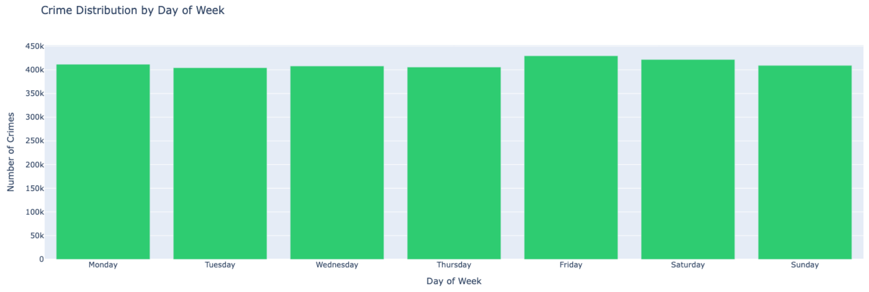
2. Crime trends over years¶



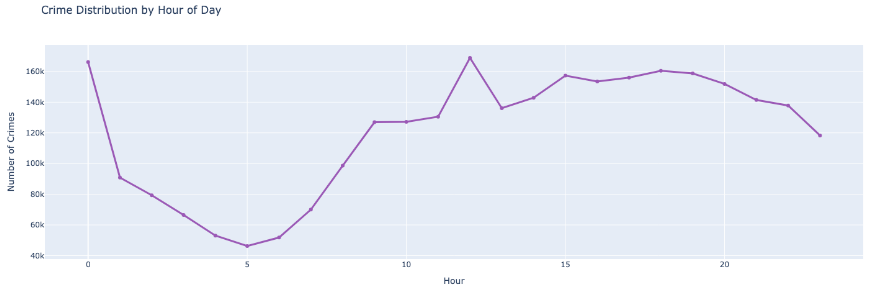
3. Crime Distribution by Month¶


Crime Distribution by Month

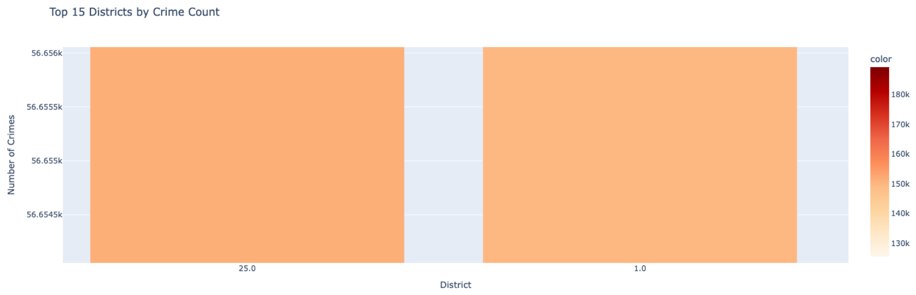
4. Crime Distribution by Week¶


Crime Distribution by Day of Week

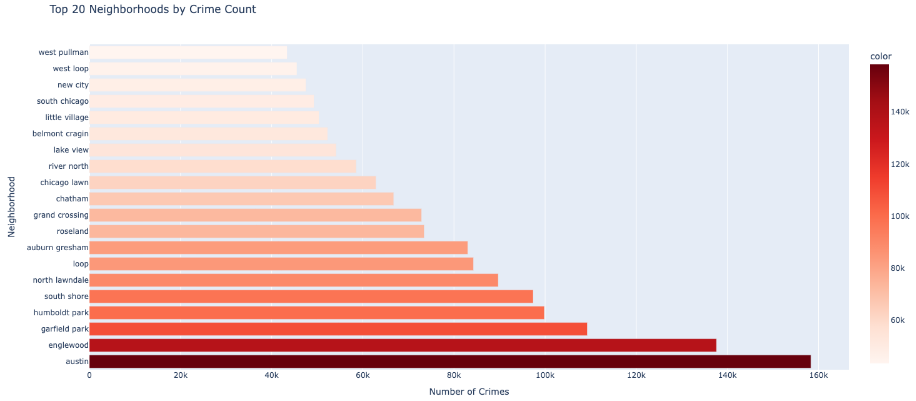
5. Crime Distribution by Hour of the Day¶


Crime Distribution by Hour of Day

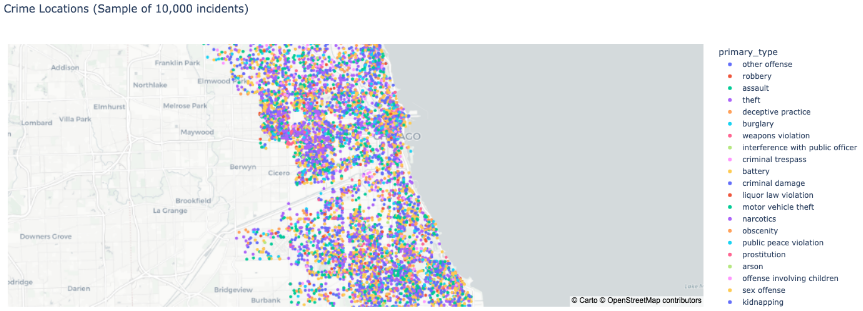
6. Top 15 Crime Districts¶



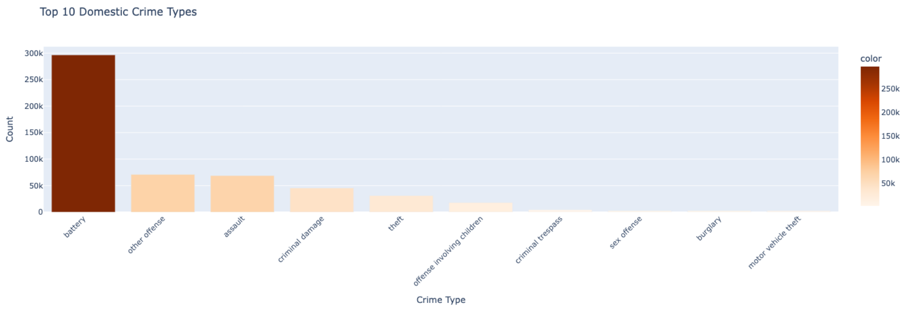
7. Top 20 Neighbourhoods by Crime¶



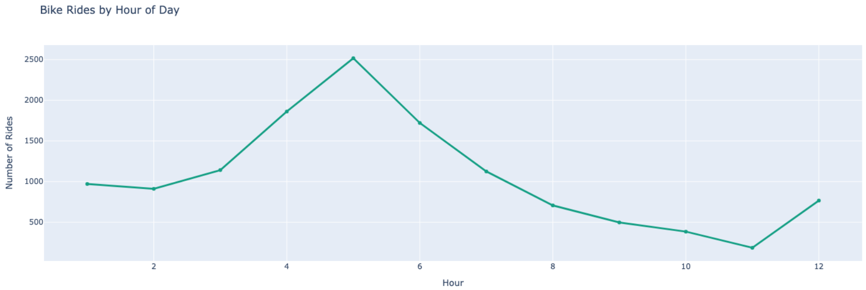
8. Crime Locations¶



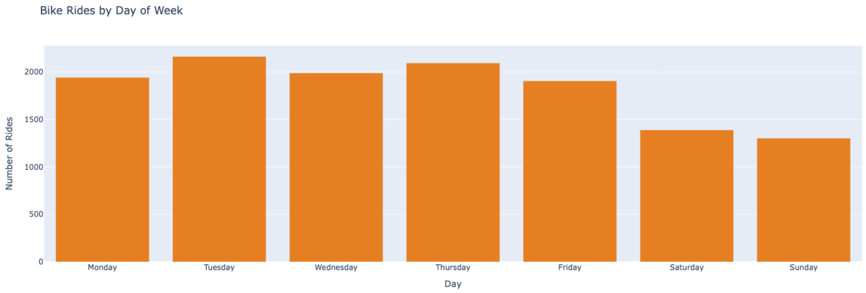
9. Top 10 Domestic Crime Types¶



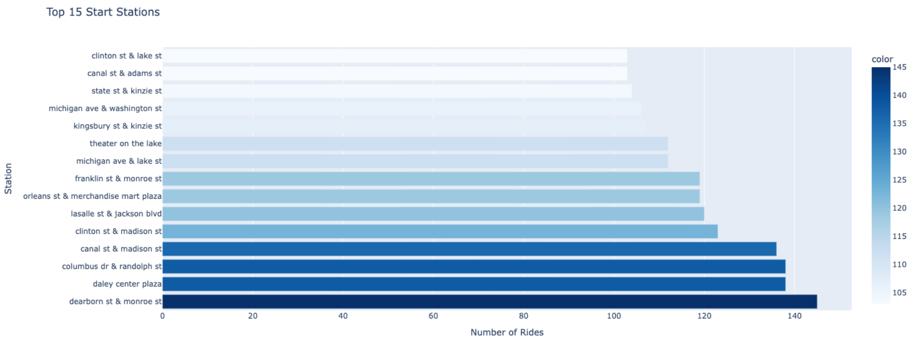
10. Bike Rides by Hour of the Day¶



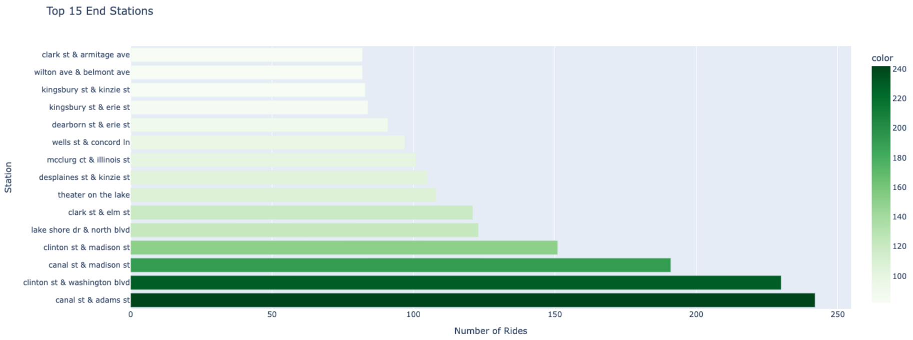
11. Bike Rides by Day of the Week¶



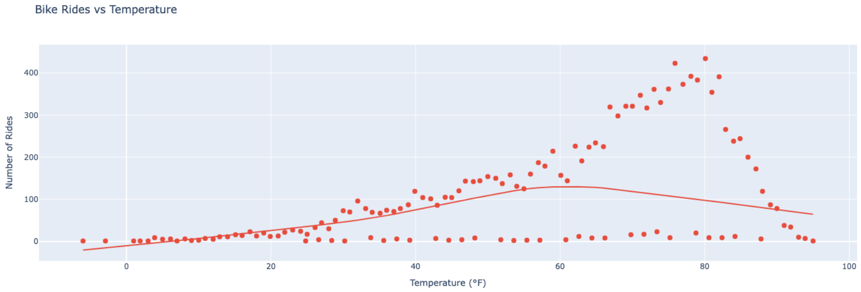
12. Top 15 Start Stations¶



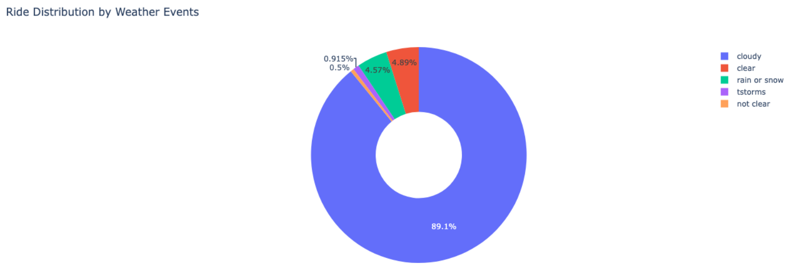
13. Top 15 End Stations¶



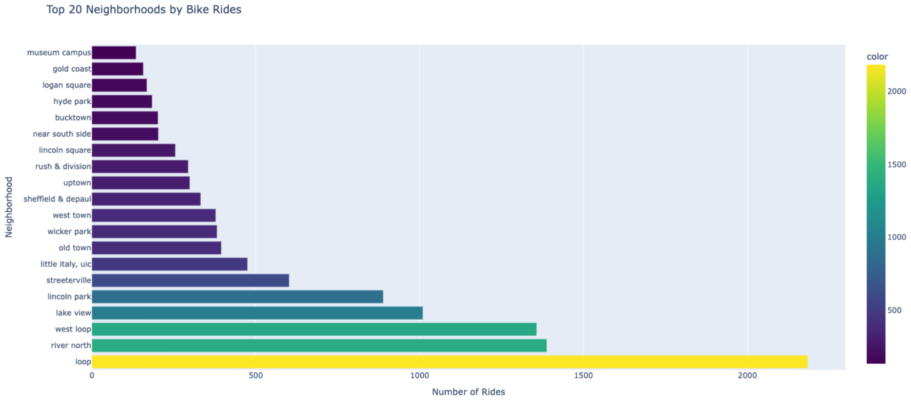
14. Bike Rides vs Temperature¶



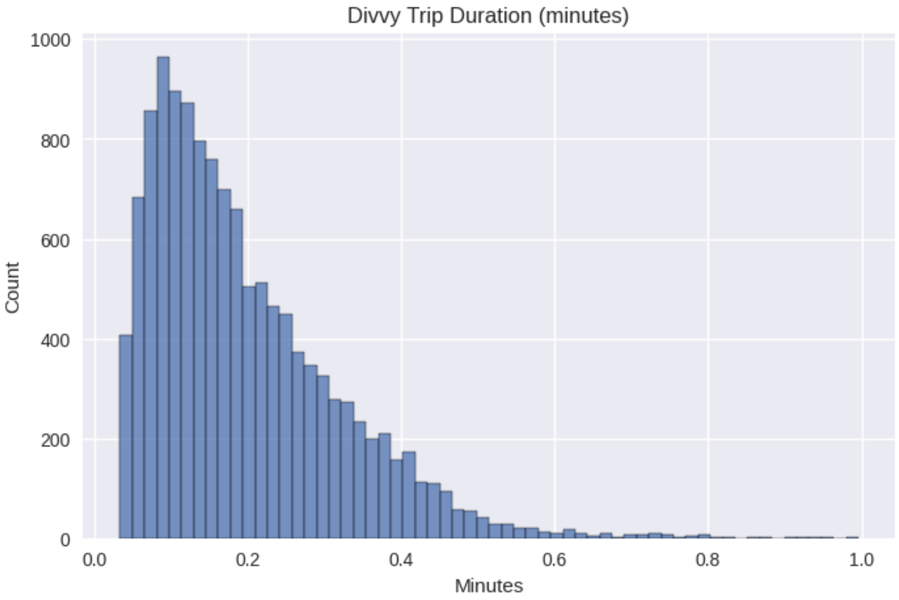
15. Ride Distribution by Weather Events¶


Ride Distribution by Weather Events

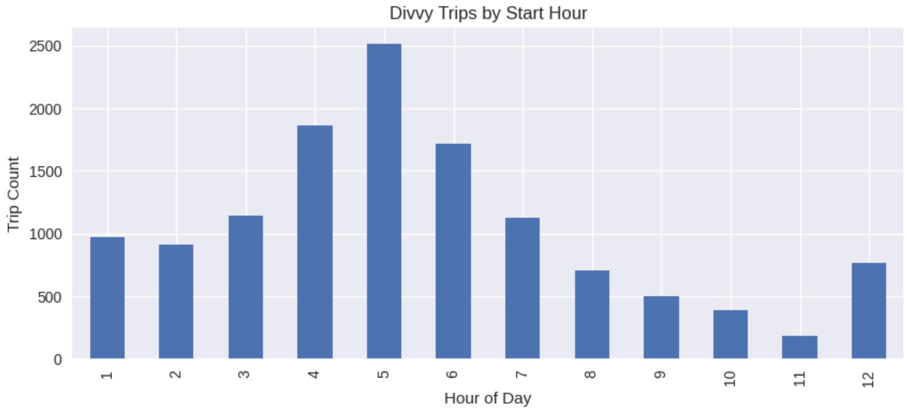
16. Top 20 Neighbourhoods by Bike Rides¶


Top 20 Neighborhoods by Bike Rides

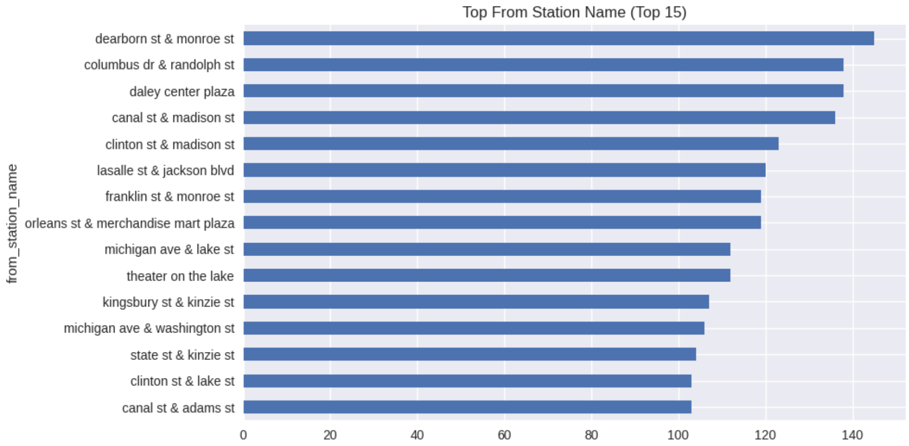
17. Divvy Trip duration distribution¶



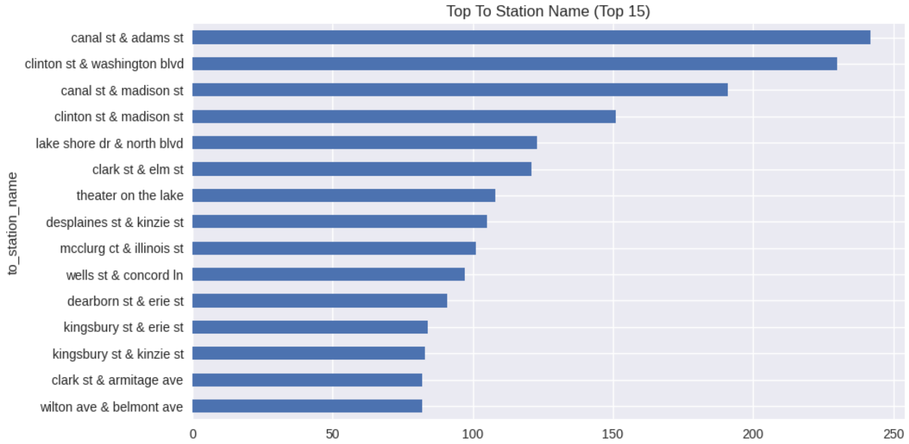
18. Peak hours¶



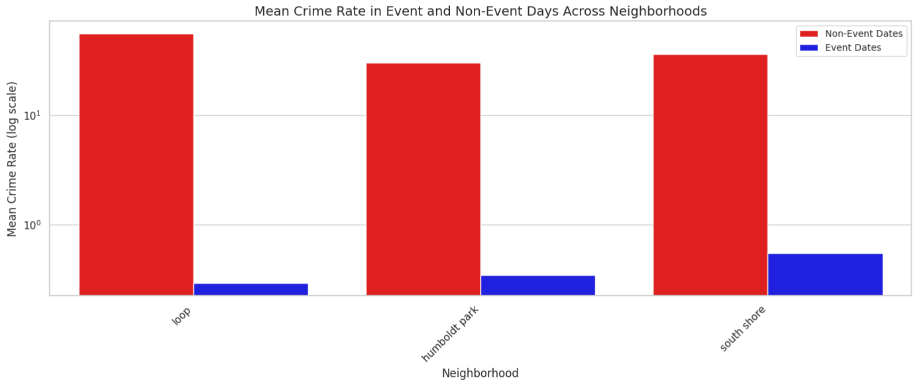
19. Top stations (from, Divvy)¶

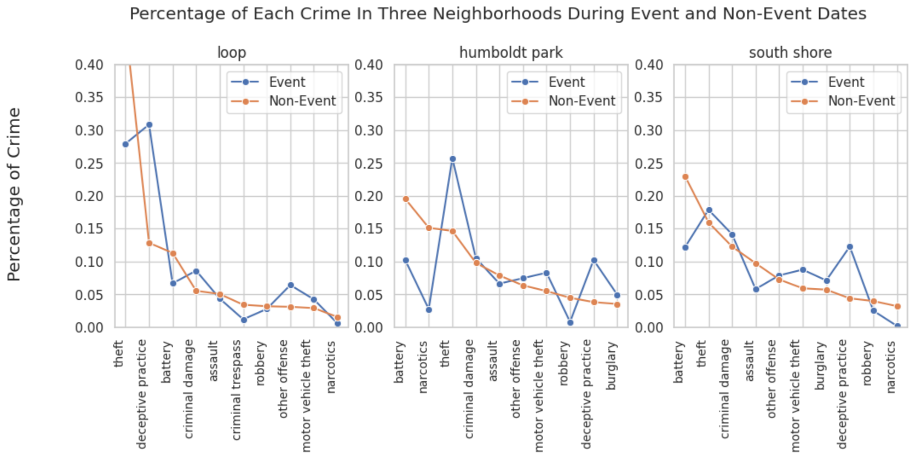


20. Top stations (to, Divvy)¶

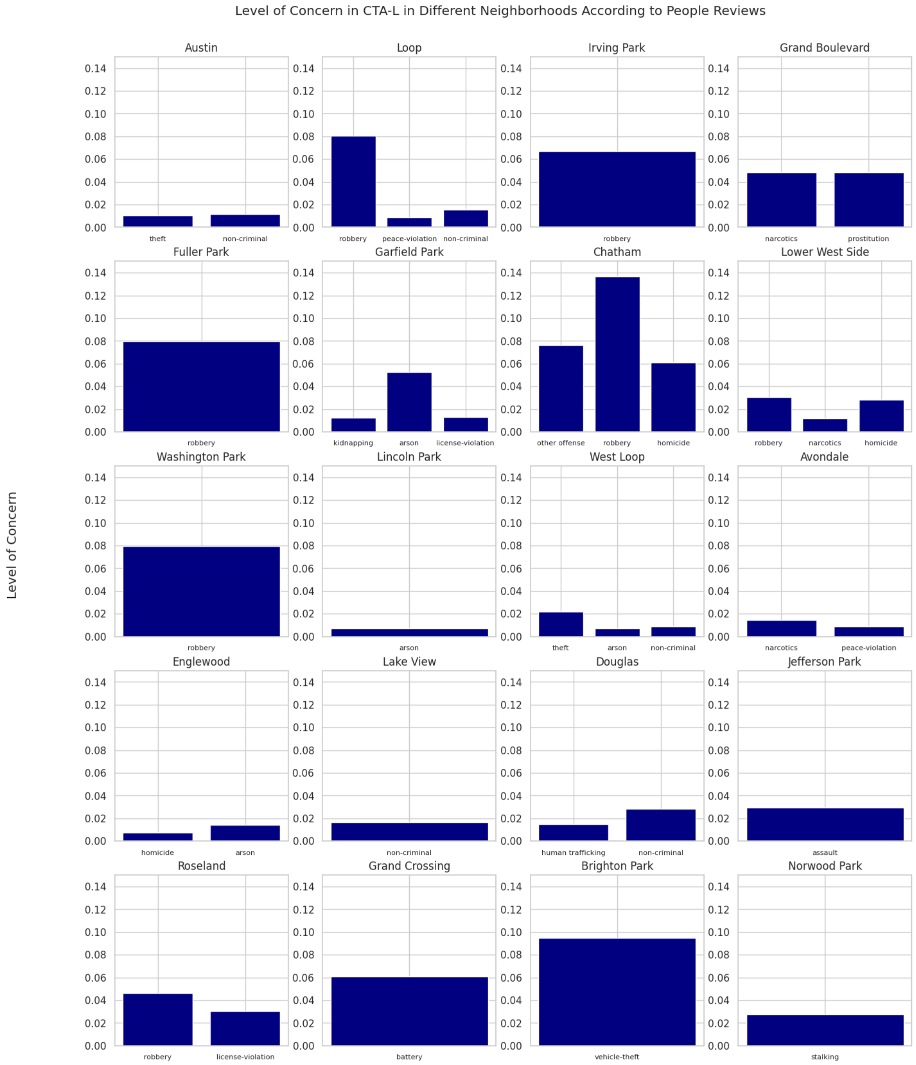


21. Crime-Event Relationship Analysis¶





22. Train Station Reviews Analysis¶



Machine Learning Models: Sentiment Classification¶

We developed two sentiment classification models using Logistic Regression with TF-IDF vectorization:

1. Park Reviews Sentiment Classifier¶

- **Task:** Classify park reviews as positive (rating > 3) or negative (rating < 3)
- **Features:** Review text using TF-IDF vectorization
- **Preprocessing:** We perform a rather similar pre-processing as in HW3. We make every token lowercased, delete apostrophes and other punctua-

tion, and delete URLs. Due to severe class imbalance, we use macro-f1 as the main evaluation.

- **Model:** Logistic Regression with balanced class weights
- **Results:**
  - **Precision (Negative):** 0.55
  - **Recall (Negative):** 0.71
  - **Macro-F1 (Negative):** 0.62
  - **Precision (Positive):** 0.98
  - **Recall (Positive):** 0.95
  - **Macro-F1 (Positive):** 0.97
  - **Macro-F1 :** .79

2. Train Station Reviews Sentiment Classifier¶

- **Goal:** Estimate the level of concern regarding different crimes in CTA-L in different neighborhoods
- **Task:** Classify train station reviews as positive (rating > 3) or negative (rating < 3)
- **Features:** Review text using TF-IDF vectorization
- **Preprocessing:** We perform a rather similar pre-processing as in HW3. We make every token lowercased, delete apostrophes and other punctuation, and delete URLs. We do not lemmatize since we observed that some subcrimes get deleted, and to keep things simple, here we void lemmatization. Due to severe class imbalance, we use macro-f1 as the main evaluation.
- **Model:** Logistic Regression with balanced class weights
- **Results:**
  - **Precision (Negative):** 0.61
  - **Recall (Negative):** 0.61
  - **Macro-F1 (Negative):** 0.61
  - **Precision (Positive):** 0.88
  - **Recall (Positive):** 0.88
  - **Macro-F1 (Positive):** 0.88
  - **Macro-F1 :** .74

Model Implementation Details¶

- Used TF-IDF vectorization to convert text reviews to numerical features
- Applied train-test split (80-20)
- Used balanced class weights to handle class imbalance
- Evaluated using classification report with precision, recall, and F1-scores

Specification¶ We have divided the reviews to positive and negative according to their scores ”1 or 2” to as negative and ”4” and ”5” to positive. Then we use TF-IDF vectorizer to get features for these reviews. We then train a logistic regression classifier and achieved the scores as mentioned above. After training, we want to assess the importance of each crime word in reviews. However, we have **two challenges** here; **Challenge I:** the words of crimes are limited and many crimes might be commented in different words (e.g. word ”robbed” is not captured if we only consider ”robbery”). **Solution I:** to solve this we use GPT-5 using a prompt and ask it to expand our vocabulary for each crime word. The specifics are shown in our notebooks. For each crime word like ”robbery”, we get all subcrime words like ”robbed”, ”robbery”, etc. **Challenge II:** How should we compute a score for each crime? **Solution II:** One way is to get the TF-IDF score for that crime in a review. However, this does not account for the label information and how that crime affects the general opinion of people regarding train stations. Thus, affected by a method like Grad-CAM from computer vision, we calculate $word_{\text{TF-IDF \:Score}} \times word_{Logistic\: Regression\: Weight\: for \:that \:word}$. Then, we sum all these subcrime scores for each crime across all train stations in each neighborhood. This way, for each neighborhood, we extract a level of concern for each crime.

Results Analysis¶ We observe that in most neighborhoods' train stations, robbery is the most concerning problem. Homicide and arson seem to be two other concerning crimes. Specifically, we see that for the most important neighborhood in the city (Loop), robbery is a very important concern. We have to note that in our analysis, because of the sum, prevalent crimes like robbery are weighted more. But, we didn't change the calculation, since the prevelance of a crime is definitely related to its level of concern. Some neighborhoods, like Catham, Engle Wood, and Lower West Side, seem to have more dengarous train stations as people have the concern of homicide.

Crime-Event Relationship Analysis¶

We analyzed the relationship between public events and crime rates in each neighborhood:

- **Challenges:** The dangerous neighborhoods with high crime rates have few events, and neighborhoods with a large number of events have a low crime rate. This makes it challenging to assess the relationship between public events and the crime rate due to high variance. Thus, we keep only the top neighborhoods in crime and events; subsequently, we take the intersection of these neighborhoods, resulting in 3 neighborhoods ”loop”, ”humboldt park”, and ”shouth shore”.

- **Computation:** Subsequently, for each neighborhood, we compute all unique event and non-event dates, and we take the average of crime incidents in event and non-event dates in these three neighborhoods (first figure in Crime-Event Relationship Analysis). Additionally, for a fine-grained analysis, we compute the percentage of each crime in event vs non-event dates (second figure in Crime-Event Relationship Analysis). This analysis shows us what specific crimes might rise or fall due to events.

- **Figure 1 Analysis:** In the first figure, we observe that, interestingly, the rate of crime significantly drops on average during event dates. This might be due to the presence of as more strict security measures. The mean crime incidents are shown in log scale since the crime rate is much higher during event dates. The exact crime rates are shown in the table below. This is very interesting, as it is in contradiction with what we believed earlier that during event dates, crimes might soar.

Index	Neighborhood	Non-Event Dates	Event Dates
0	south shore	36.254416	0.550453
1	humboldt park	30.409355	0.345455
2	loop	56.191919	0.290710

- **Figure 2 Analysis:** In the second figure, we observe that a violent crime like battery has a higher percentage in event dates across neighborhoods compared to non-event dates. Whereas a crime like deceptive practices has a much higher rate during event dates. This is expected, as during event dates, a large group of people gathers, and the conditions for such actions are more suitable. Criminal damage has a bit higher percentage during event dates, since there is probably more communication and incidents are more likely to happen. Theft experiences an inconsistent shift in different neighborhoods. Interestingly, theft has a large increase in proportion compared to non-event dates, probably because the loop is very busy and the conditions for theft become very suitable as compared to South Shore and Humboldt Park, which are relatively less busy areas.

3. Divvy Bike Sharing Dataset Analysis¶

Dataset Specifications¶

- **Size:** 19,899 trips (12,791 after cleaning)
- **Time Period:** Historical bike sharing data
- **Key Features:**
  - Trip start and end times
  - Trip duration (in seconds/minutes)
  - Start and end station information
  - Geographic coordinates (start and end)
  - Neighborhood assignments (start and end)
  - Temperature data
  - Weather events
  - Station capacity information

Data Cleaning¶

- Removed redundant index columns
- Converted timestamp columns to datetime format

- Removed rows with invalid timestamps
- Standardized column names
- Created temporal features (year, month, hour, day of week)
- Calculated trip duration in minutes

Key Findings¶

- Trip patterns vary by time of day and day of week
- Weather conditions affect bike sharing usage
- Popular routes between neighborhoods
- Station capacity utilization patterns

Divvy Visualizations¶

Note: Divvy visualizations are included in the Crime & Divvy EDA notebook. See `datasets/Crime/EDA/Crime_and_Divvy_EDA.ipynb` for detailed Divvy bike sharing analysis and visualizations. The Divvy dataset was analyzed alongside crime data to examine transportation patterns and their relationship with urban dynamics.

4. Events Dataset Analysis¶

Dataset Specifications¶

- **Size:** 133,799 event permits
- **Time Period:** 2012-2025
- **Key Features:**
  - Park/Facility name
  - Event type
  - Reservation start and end dates
  - Permit status
  - Geocoded coordinates (latitude, longitude)
  - Neighborhood assignments

Data Processing Steps¶

1. **Geocoding:** Converted park/facility names to coordinates using:
   - Nominatim geocoder (reverse geocoding)
   - OpenAI API for ambiguous locations
   - Manual mapping for special cases
2. **Neighborhood Mapping:** Assigned events to Chicago neighborhoods using coordinates
3. **Temporal Analysis:** Extracted year, month, and date features

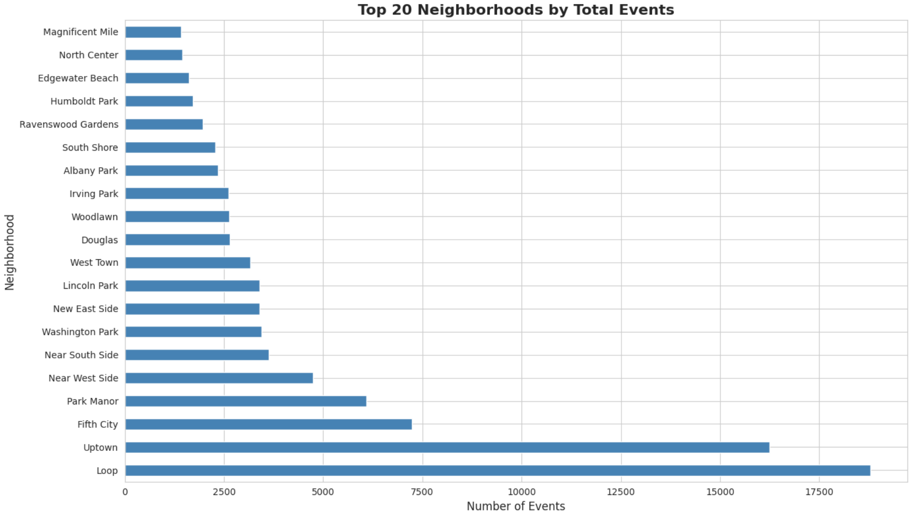
Key Findings¶

- Top neighborhoods by event count: Loop, Uptown, Fifth City, Park Manor

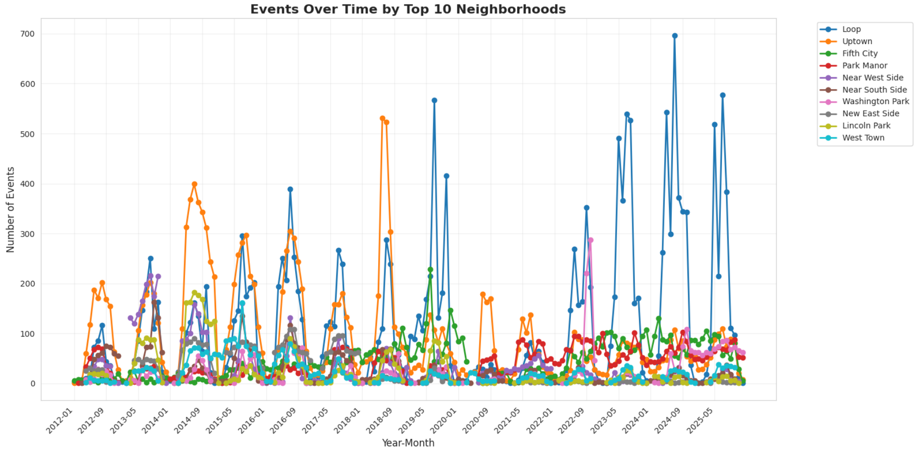
- 187 unique neighborhoods with events
- Temporal patterns show seasonal variations
- Event types vary by neighborhood
- 98.6% of events successfully geocoded and mapped to neighborhoods

Events Visualizations¶

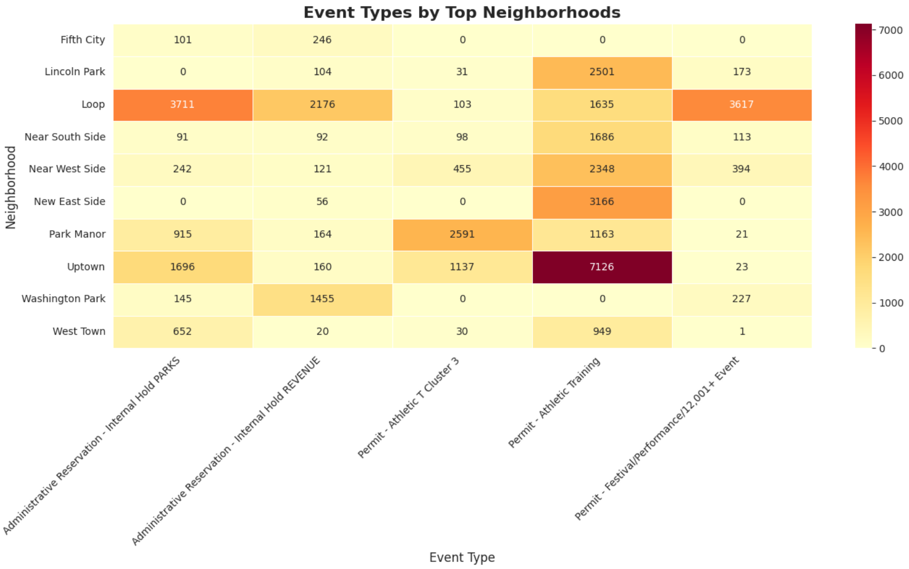
1. Top 20 Neighborhoods by Total Events¶



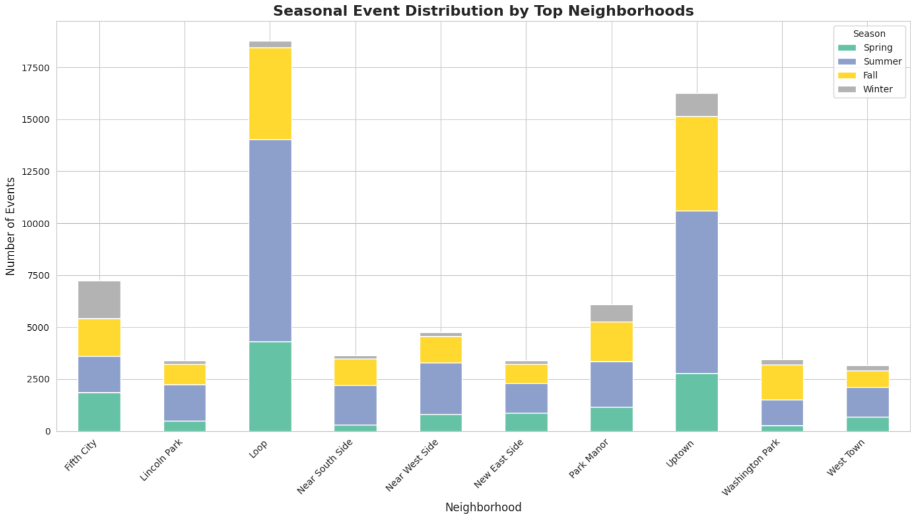
2. Events Over Time by Top 10 Neighborhoods¶



3. Event Types by Top Neighborhoods (Heatmap)¶



4. Seasonal Event Distribution by Top Neighborhoods¶



5. Animated Visualizations¶ The following animated GIFs show temporal patterns in event distribution:

- **Monthly Events by Neighborhood**: `datasets/Events/chicago_events_by_neighborhood_animate`
  - Shows events per month for top 20 neighborhoods
- **Cumulative Events Over Time**: `datasets/Events/chicago_events_cumulative_animated.gif`
  - Shows cumulative events by neighborhood over time

7. Interactive Maps¶ Interactive HTML maps are available:

- **Event Locations Map**: `datasets/Events/chicago_events_map.html` - Folium map with event markers and popup information
- **Event Density Heatmap**: `datasets/Events/chicago_events_heatmap.html` - Density heatmap showing concentration of events across Chicago

Geocoding Process¶

The geocoding process involved:

1. **Initial Attempt:** Direct geocoding of park/facility names
2. **Fallback Methods:**
   - Adding ”Chicago, IL” to location names
   - Using OpenAI API for ambiguous locations
   - Manual verification for special cases
3. **Results:**
   - Successfully geocoded 99.66% of events
   - Saved geocoding mapping to `park_facility_coordinates.json`
   - Includes latitude, longitude, query used, and geocoding method

5. Weather Dataset Analysis¶

Dataset Specifications¶

- **Size:** 974 records (after cleaning)
- **Time Period:** January 1, 2023 to August 31, 2025
- **Data Source:** NOAA API (Chicago O'Hare station)
- **Key Features:**
  - Temperature (max, min, average, feels like)
  - Precipitation (amount, probability, type)
  - Wind (speed, direction, gusts)
  - Humidity
  - Cloud cover
  - Visibility
  - Solar radiation
  - UV index
  - Sea level pressure
  - Weather conditions

Data Collection¶

- Fetched data from NOAA CDO API
- Station: GHCND:USW00094846 (Chicago O'Hare)
- Retrieved 13,019 records across multiple data types
- Pivoted data to have dates as rows and data types as columns

Data Cleaning¶

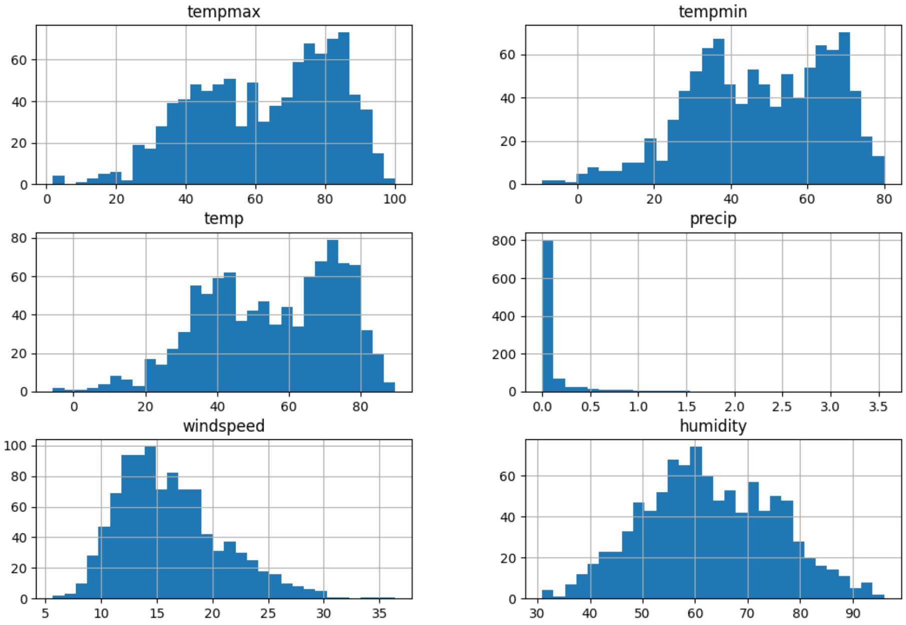
- Converted the datetime column to proper format
- Removed irrelevant columns (stations, description, icon, name)
- Interpolated missing numeric values using linear interpolation
- Filled categorical columns with mode (most frequent value)
- Created temporal features (month, year, day of week, is_weekend)

Key Findings¶

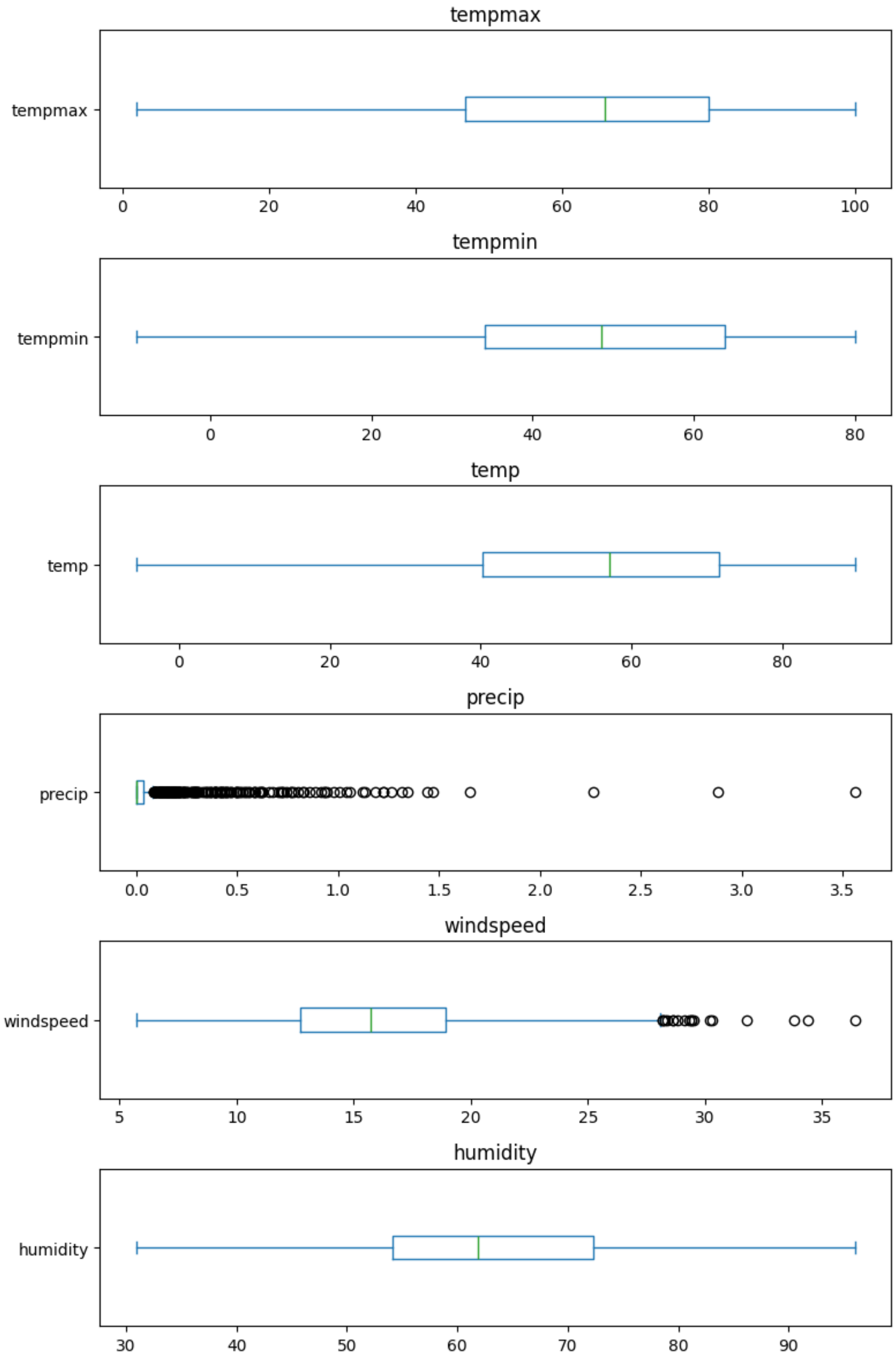
- Complete weather data for analysis period
- Seasonal temperature patterns
- Precipitation patterns and types
- Wind patterns throughout the year

Weather Visualizations¶

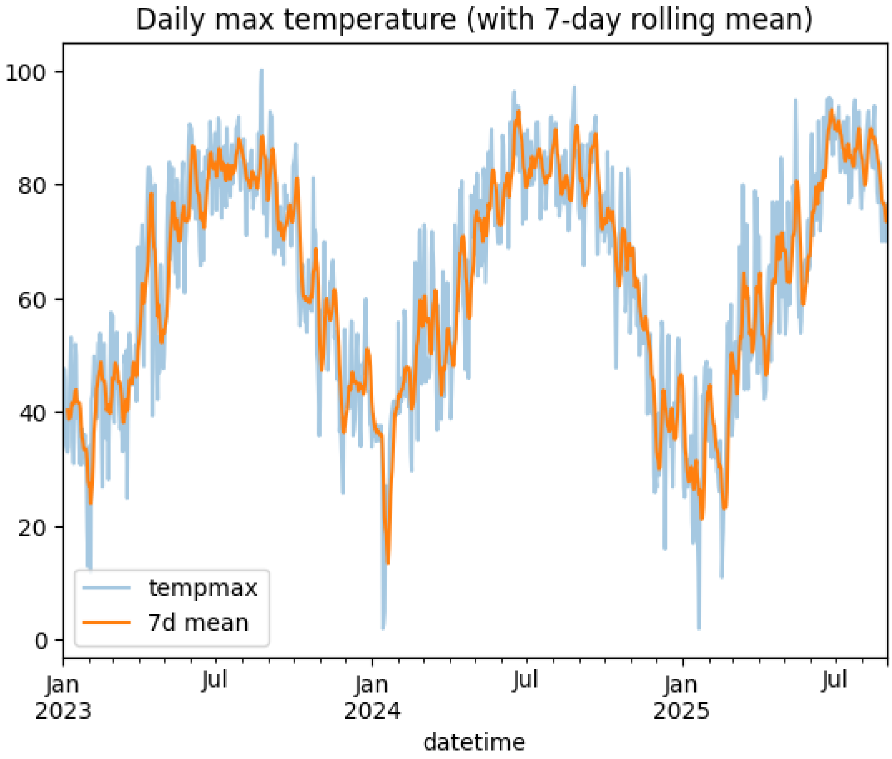
1. Weather Variable Histograms¶



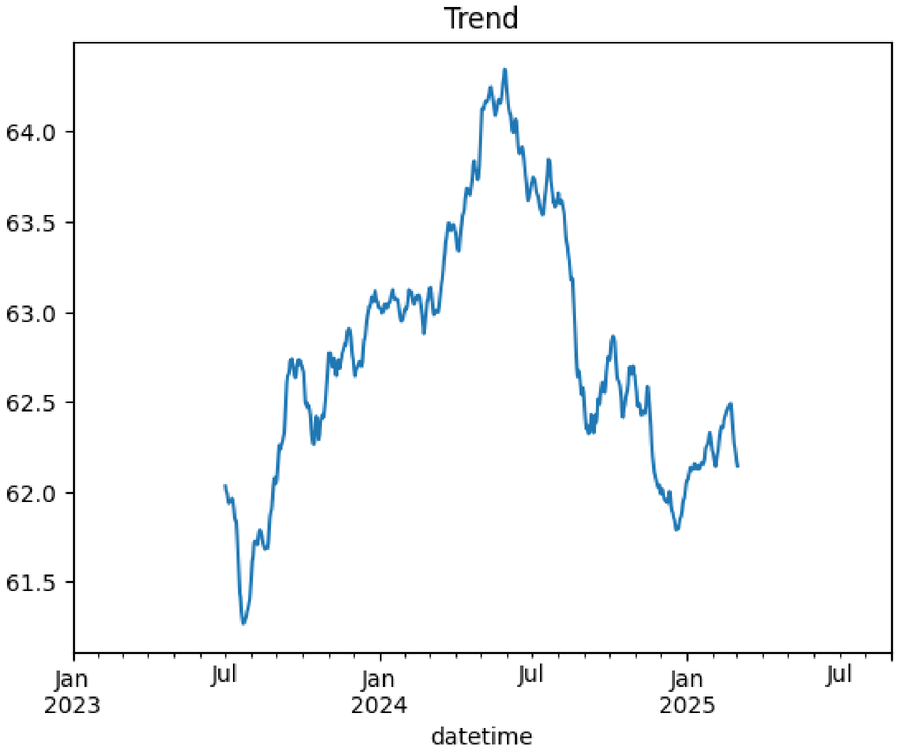
2. Weather Variable Boxplots¶

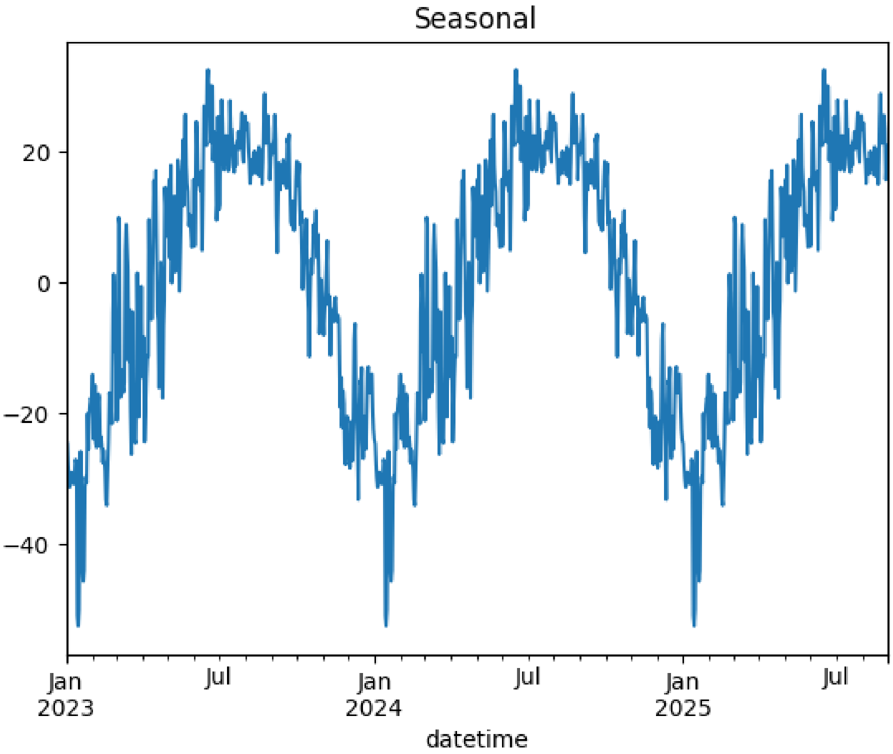


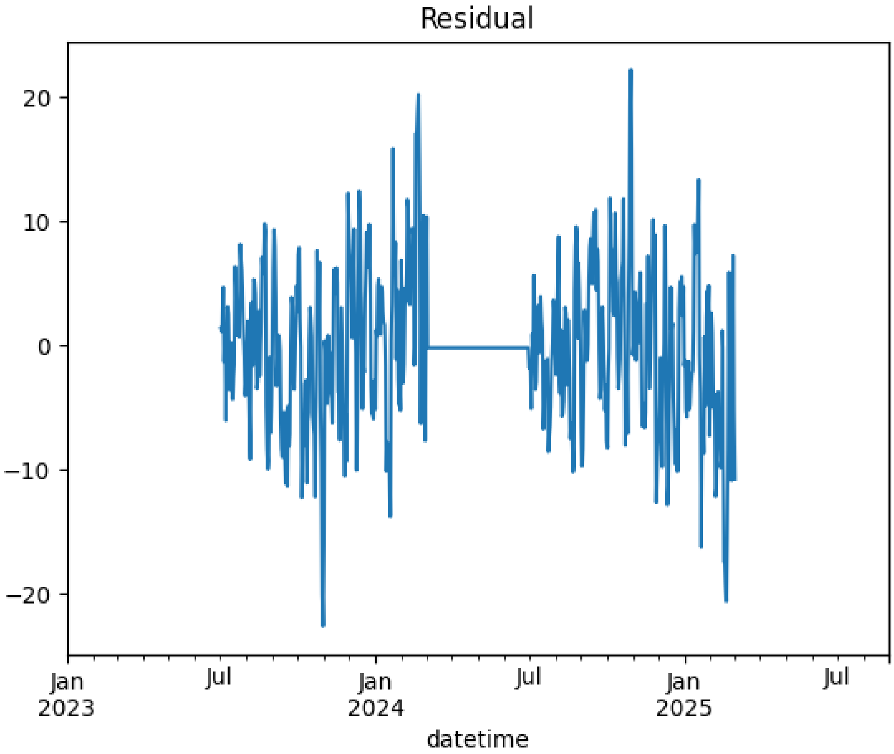
3. Daily Maximum Temperature with 7-Day Rolling Mean¶



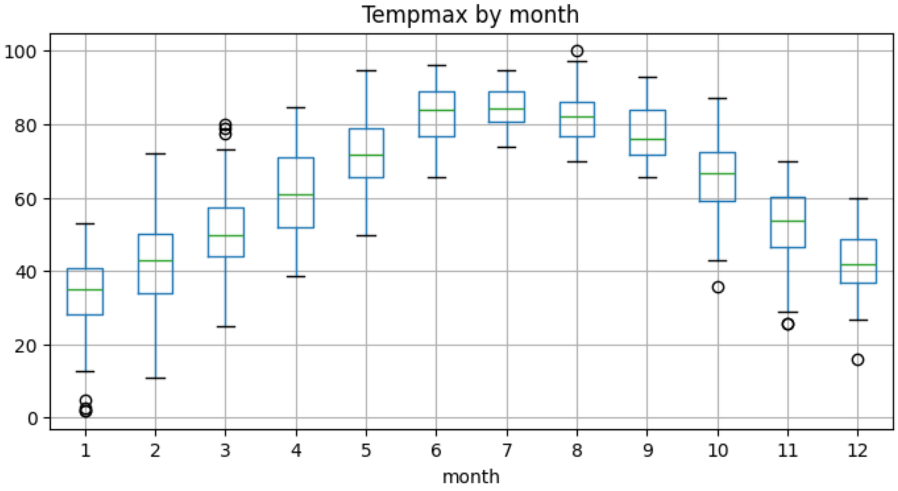
4. Temperature Trend Decomposition¶


Trend
64.0
63.5
63.0
62.5
62.0
61.5
Jan
2023
Jul
Jan
2024
Jul
Jan
2025
Jul
datetime

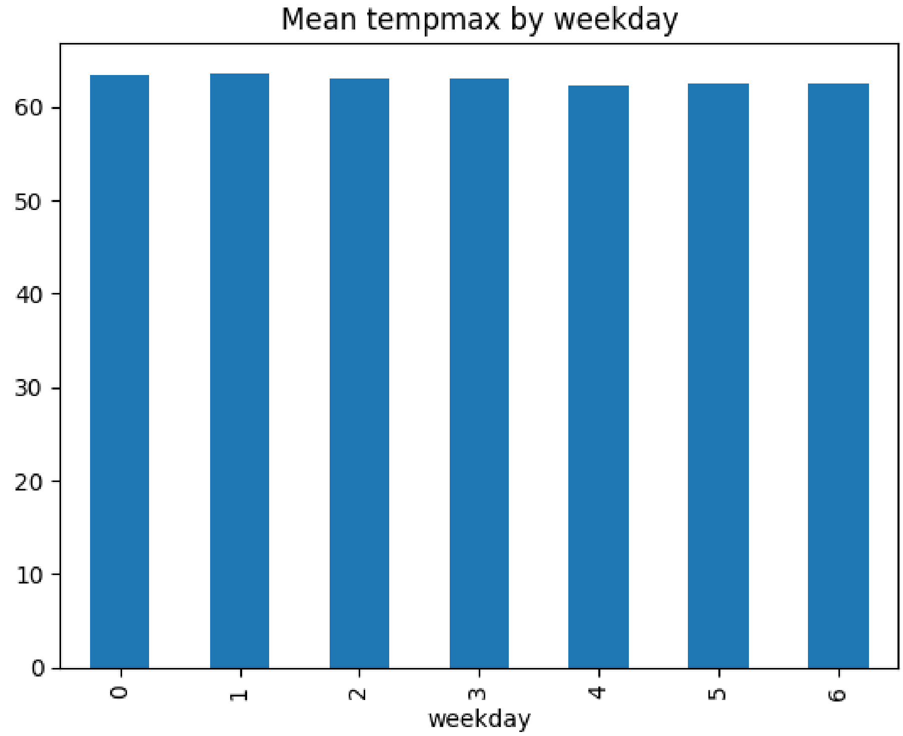

Seasonal
20
0
−20
−40
Jan
2023
Jul
Jan
2024
Jul
Jan
2025
Jul
datetime



5. Maximum Temperature by Month¶



6. Mean Maximum Temperature by Weekday¶



Crime–Weather Dataset Merge¶

The geocoding process involved:

1. **Preparation:**
   - Converted crime timestamps to daily format
   - Aggregated total crimes per date (city-wide)
2. **Analysis Performed:**
   - Only date-based merging performed
   - Compared daily crime counts across:
     - Rain vs. no rain
     - Temperature ranges etc.
   - Generated simple correlations and trend plots
3. **Key Findings:**
   - **Rainy days:** lower outdoor crime levels
   - **Snow days:** significant drop in mobility-related crimes

- **Warm days:** slight increase in disorderly conduct and high-activity crime types
- Weather effects are most visible for outdoor crime categories

4. **Output:**

   - Created merged table: `weather_crime_merged.csv`
   - Produced comparison plots for temperature, precipitation, and snow days
   - Ready for downstream modelling and seasonal trend analysis

Weather-Crime Visualisations¶

Note: The visualisations are included in the Weather and Crime Relations notebook. See `datasets/Weather/Weather_Crime_Relations.ipynb` for detailed analysis and visualisations. Temperature shows the strongest relationship, with crime increasing on hot days and in summer months. Precipitation and snow are associated with modest declines in crime, suggesting reduced outdoor activity. Together, the plots illustrate how daily and seasonal weather shifts impact overall crime patterns.

6. Preprocessing: Park and Train Station Reviews¶

Data Collection¶

We collected reviews for:

1. **Chicago Parks:** Using Google Places API
2. **CTA Train Stations:** Using Google Places API

Process¶

1. **Station/Park Identification:**

   - Used Google Places Nearby Search API
   - Searched for train stations and parks near known coordinates
   - Retrieved place IDs for each location

2. **Review Collection:**

   - Used Google Places Details API
   - Retrieved reviews, ratings, and metadata
   - Stored in JSON format

3. **Data Structure:**

   - Each park/station has:
     - Overall rating
     - Number of raters
     - List of reviews (text, rating, time)

Results¶

- Collected reviews for 145 CTA train stations
- Collected reviews for multiple Chicago parks
- Used for sentiment analysis and crime correlation

Note: See `datasets/Preprocessing/Get_Park_Train_Reviews.ipynb` for implementation details

7. Integration and Cross-Dataset Analysis¶

Crime-Event Integration¶

- Analyzed crime rates on event days vs non-event days
- Found neighborhoods with both high crime and high event activity
- Compared crime type distributions during event and non-event days

Crime-Divvy Integration¶

- Analyzed bike sharing patterns in high-crime areas
- Examined transportation patterns relative to crime hotspots

Weather-Transportation Integration¶

- Divvy dataset includes temperature data
- Can correlate weather with bike sharing usage

Neighborhood-Based Analysis¶

- All datasets mapped to Chicago neighborhoods
- Enables cross-dataset neighborhood-level analysis
- Supports spatial correlation studies

8. Technical Specifications¶

Tools and Libraries Used¶

- **Data Processing:** pandas, numpy
- **Visualization:** matplotlib, seaborn, plotly, folium
- **Geocoding:** geopy, OpenAI API
- **Machine Learning:** scikit-learn (TF-IDF, Logistic Regression)
- **Animation:** imageio, matplotlib.animation
- **APIs:** Google Places API, NOAA CDO API

Data Storage¶

- CSV files for processed datasets
- JSON files for geocoding mappings and reviews
- HTML files for interactive maps

- GIF files for animated visualizations

File Structure¶

```
datasets/
  Crime/
     crime_with_neighborhoods.csv
     EDA/
        Crime_and_Divvy_EDA.ipynb
        Crime_And_Event_EDA_W_ML_Models.ipynb
  Divvy/
     divvy_sample_with_neighborhoods.csv
  Events/
     Chicago_Park_District_-_Event_Permits_With_Neighborhoods.csv
     chicago_events_map.html
     chicago_events_heatmap.html
     chicago_events_by_neighborhood_animated.gif
     chicago_events_cumulative_animated.gif
  Weather/
     Chicago_weather_cleaned.csv
Preprocessing
   Cleaning_Crime_Event_Divvy.ipynb
```

9. Challenges and Solutions¶

Challenge 1: Geocoding Park/Facility Names¶

Problem: Many park/facility names were ambiguous or incomplete

Solution:

- Implemented multiple fallback strategies
- Used OpenAI API for intelligent location resolution
- Created manual mapping for special cases
- Achieved 99.66% geocoding success rate

Challenge 2: Large Dataset Processing¶

Problem: Crime dataset with 2.9M records required efficient processing

Solution:

- Used efficient pandas operations
- Implemented sampling for visualization
- Optimized memory usage

Challenge 3: Class Imbalance in Sentiment Classification¶

Problem: Positive reviews significantly outnumbered negative reviews

Solution:

- Used balanced class weights in Logistic Regression
- Achieved good performance on both classes

Challenge 4: Temporal Data Alignment¶

Problem: Different datasets had different temporal granularities

Solution:

- Standardized date formats
- Created common temporal features
- Aligned data by date for cross-dataset analysis

Challenge 5: Limited Crime Vocabulary¶

Problem: The lack of crime vocabulary diversity in our crime dataset

Solution:

- Used GPT-5 to expand our vocabulary for each crime, and got many subcrime words for that crime

Challenge 6: Extracting Neighborhoods¶

Problem: Extracting neighborhoods given each coordinate

Solution:

- Received ideas from the public repository `https://github.com/craigmbooth/chicago_neighborhood_f` to extract Chicago neighborhoods given coordinates

Challenge 6: Restricted Reviews¶

Problem: The Limitation of the Google API to returns only 5 reviews for each place

Solution:

- For our final result, we seek to retrieve nearby places in each neighborhood to get more data.
- Using pre-trained word embeddings to initialize word vectors for a better result
- Using context-based models like Bert to get better performance.

10. Results Summary¶

Data Processing Results¶

- **Crime Dataset:** 2.9M records processed and analyzed
- **Divvy Dataset:** 12.8K trips processed after cleaning

- **Events Dataset:** 133.8K events geocoded and mapped to neighborhoods
- **Weather Dataset:** 974 days of weather data collected and cleaned

Machine Learning Results¶

- **Park Reviews Classifier:** 79% Macro-F1
- **Train Station Reviews Classifier:** 74% Macro-F1

Visualization Results¶

- Interactive maps for event locations
- Heatmaps for event density
- Animated GIFs for temporal patterns
- Multiple static visualizations for all datasets

Analysis Results¶

- Identified top crime types and patterns
- Mapped events to 187 neighborhoods
- Analyzed crime-event relationships
- Identified temporal patterns across all datasets

11. Next Steps and Future Work¶

Immediate Next Steps¶

1. **Enhanced Predictive Models:**
   - Develop crime prediction models using multiple features
   - Incorporate weather and event data into predictions
   - Explore time series forecasting models
2. **Advanced Analysis:**
   - Deep dive into crime-event correlations
   - Analyze transportation patterns in relation to safety
   - Weather impact on urban activities
3. **Model Improvements:**
   - Experiment with different ML algorithms
   - Feature engineering for better predictions
   - Hyperparameter tuning
4. **Visualization Enhancements:**
   - Interactive dashboards
   - Real-time data visualization
   - More sophisticated temporal visualizations

Long-term Goals¶

1. Real-time urban monitoring system
2. Predictive analytics dashboard
3. Integration with additional data sources
4. Deployment of models for production use

12. Conclusion¶

We have successfully completed Phase 3 of the UrbanPulse project with significant progress across all datasets. Our work includes:

- Comprehensive data collection and preprocessing
- Extensive exploratory data analysis
- Successful geocoding and neighborhood mapping
- Development of machine learning models with good performance
- Creation of informative visualizations including animated GIFs
- Cross-dataset integration and analysis

The project demonstrates the value of integrating multiple urban data sources to gain insights into city dynamics. Our models and analyses provide a foundation for predictive urban analytics and informed decision-making.

Report Prepared By: UrbanPulse Team

Date: 11/14/2025