Problem Statement

Cloudbursts are sudden, extreme rainfall events—typically exceeding 100 mm/hour within a localized area—that overwhelm natural and built infrastructure, often triggering flash floods and landslides. These events are most common in mountainous or urban regions where topography or impermeable surfaces exacerbate runoff. Meteorologically, cloudbursts are associated with convective storms, high atmospheric moisture content, and unstable atmospheric conditions. Despite their intensity and frequency in certain regions, forecasting them remains difficult due to their small spatial and temporal scales.

The socioeconomic consequences of cloudbursts can be devastating. In urban areas, they can lead to infrastructure damage, business disruption, casualties, and loss of life—particularly in low-income communities with inadequate drainage systems. For municipalities, businesses, and disaster relief agencies, the ability to anticipate such events even a few hours in advance could dramatically improve emergency response and mitigation efforts.

However, cloudburst prediction poses significant technical challenges. Their rarity and highly localized nature mean that many weather datasets lack the spatial and temporal resolution required to capture them. Moreover, traditional statistical methods often fail to achieve meaningful prediction accuracy. Machine learning offers promise, but requires extensive, high-quality data and careful model design.

The INDRA project, developed in partnership with Alt Surya Inc., aims to address this gap by building a machine learning-based early warning system that can predict cloudbursts within a three-hour window. The project's primary objectives include improving prediction accuracy over baseline models, estimating impact severity, and producing actionable alerts that support timely decision-making in urban disaster preparedness.

Literature Review

1. A Hybrid Machine Learning–Numerical Weather Prediction Approach for Rainfall Prediction - Patil, A. A., and K. Kulkarni, 2023

- Data & Preprocessing: Used GHCN rainfall data from 55 Indian stations (June–September 2015). Extracted 50 atmospheric features from the NCEP GFS forecasts, including humidity, pressure, cloud water mixing ratios, etc., at multiple vertical layers.
- ML Methods: Compared binary classification, multiclass classification, and regression using Linear SVM, Neural Networks, and XGBoost. Used both nowcasts and 6-hourly forecasts up to 7 days.
- Key Results: The hybrid ML+GFS model achieved up to 25% improvement in F1-score over the GFS-only baseline. XGBoost performed best for multiclass, NeuralNet for regression. Forecasts (FS-2) outperformed nowcasts (FS-1).
- Limitations: Performance declines significantly after 2 days; class imbalance in rainfall data required careful metric selection (F1-score over accuracy).

2. Cloud Burst Prediction System using Machine Learning - Herin Shani, S., and G. Nagappan, 2024

- Data & Preprocessing: Integrated meteorological parameters (temperature, humidity, UV index, cloud cover) and transformed time-series data into Gramian Angular Fields (GAFs) for CNN analysis.
- ML Methods: Employed CNNs to extract spatial-temporal features from GAF-transformed data. Benchmarked against CatBoost, Random Forest, Logistic Regression, and XGBoost.
- Key Results: CNN achieved 86.42% accuracy with an F1-score of 0.93 for non-cloudburst events and 0.66 for cloudburst events. CNN outperformed traditional classifiers.
- Limitations: Lower recall for actual cloudburst cases (i.e., high false negatives), making real-world early warning applications riskier without improvement.

3. Kerala Floods in Consecutive Years—Its Association with Mesoscale Cloudbursts and Structural Changes in Monsoon Clouds - Vijaykumar, P., and Coauthors, 2021

- Data & Preprocessing: Analyzed Kerala flood events using IMDAA and ERA5 reanalysis datasets. Focused on mesoscale convective systems, ocean-atmospheric coupling, and vertical wind shear.
- ML/Analytical Approach: While not ML-based, the study provides important climatological context using mesoscale meteorology to explain cloudburst precursors and sea temperature anomalies.
- Key Results: Found that warm sea surface temperatures and vertical instability significantly contributed to unusual convective intensities in 2018 and 2019. IMDAA outperformed ERA5 in resolving small-scale precipitation bursts.
- Limitations: No predictive modeling conducted; work is observational but critical for feature selection in future ML models.

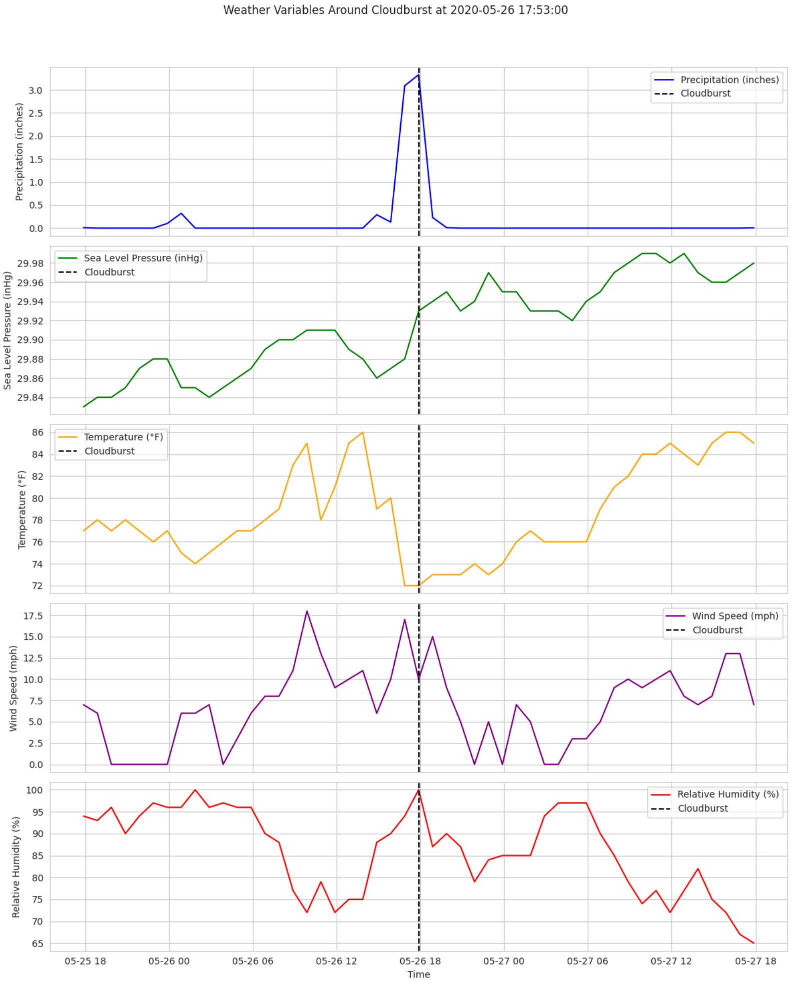
The current literature on cloudburst and extreme precipitation prediction offers valuable guidance for developing INDRA's predictive system. Patil and Kulkarni (2023) present a hybrid approach that combines Global Forecast System (GFS) outputs with machine learning models, demonstrating that incorporating both atmospheric nowcasts and forecasts significantly improves prediction accuracy. Their method—evaluating binary, multiclass, and regression tasks using models like XGBoost and neural networks—achieved up to 25% improvement over baseline forecasts, emphasizing the benefit of blending physical and data-driven features. Complementing this, Shani and Nagappan (2024) employ Gramian Angular Fields (GAF) to convert time-series weather data into spatial representations analyzed by Convolutional Neural Networks (CNNs). This method reached an overall accuracy of 86.4%, showcasing CNN's effectiveness in capturing complex meteorological patterns, though it struggled with recall in predicting actual cloudbursts—highlighting the importance of addressing class imbalance. Finally, Vijaykumar et al. (2021) contribute essential climatological insights by linking mesoscale atmospheric dynamics and warm oceanic conditions to cloudburst activity during Kerala's 2018–2019 floods, underscoring the need to account for vertical instability and sea surface temperatures in feature selection. Together, these studies support INDRA's strategy of fusing high-resolution meteorological inputs, domain-informed feature engineering, and advanced ML architectures to enhance short-term cloudburst forecasting and risk mitigation in urban settings.

Data Processing and Feature Engineering

The NOAA Local Climatological Data (LCD) dataset provided extensive high-resolution hourly observations from 2015 to 2024 at a Florida weather station. The primary variables included temperature, dew point, humidity, sea-level pressure, wind speed, wind direction, and hourly precipitation. The original dataset encompassed approximately 87,650 records, presenting substantial data preprocessing challenges. Standardizing timestamps, addressing inconsistencies, converting non-numeric entries, and filling missing data were essential steps taken to ensure dataset quality and usability for modeling.

Initially, an exploratory analysis identified a cloudburst threshold of 0.41 inches/hour (10.41 mm/hour), corresponding to the minimum precipitation event classified as heavy rainfall. This threshold revealed a severe class imbalance, as only 0.49% of records qualified as heavy rainfall events, emphasizing the rarity and critical nature of these occurrences. Visualization of precipitation data distributions using logarithmic scaling offered clear insights into extreme event patterns. Detailed inspection of a prominent cloudburst event on May 26, 2020, highlighted rapid atmospheric changes such as a wind speed surge to 15 mph, relative humidity peaks near 90%, a notable pressure drop of 0.02 inHg, and a swift temperature decline from 80°F to 74°F, aligning closely with known meteorological precursors.

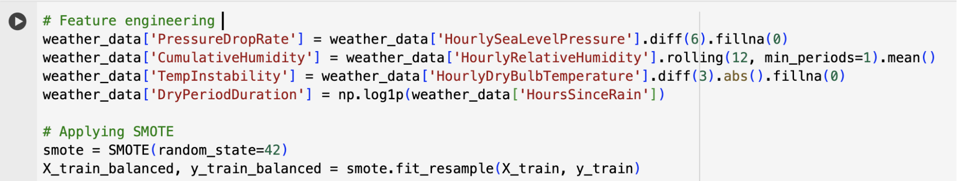
Highlighted Cloudburst Event on May 26, 2020



Guided by meteorological literature and leveraging insights from AI-driven exploratory analyses, several engineered features were created to encapsulate relevant atmospheric conditions preceding cloudbursts. These included a pressure drop rate (three-hour differential), cumulative humidity (12-hour rolling average), temperature instability (three-hour absolute temperature differential), and an indicator of monsoon wind surge events based on upper-quartile wind speeds. A significant addition was the DryPeriodDuration, calculated as the log-transformed duration of rainless hours, capturing critical atmospheric conditioning effects before heavy rainfall.

Temporal integrity and class imbalance were prominent challenges addressed through methodological decisions. Data splitting utilized weekly cluster sampling to maintain chronological coherence, thereby preventing temporal leakage. Additionally, SMOTE (Synthetic Minority Over-sampling Technique) was applied after feature engineering to rebalance the classes, enabling better model training and improved detection capability.

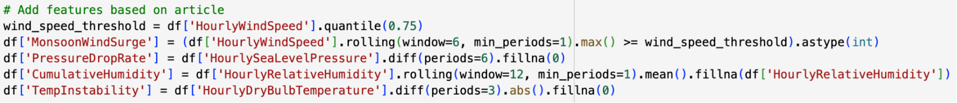
Basic Feature Engineering



```
# Feature engineering
weather_data['PressureDropRate'] = weather_data['HourlySeaLevelPressure'].diff(6).fillna(0)
weather_data['CumulativeHumidity'] = weather_data['HourlyRelativeHumidity'].rolling(12, min_periods=1).mean()
weather_data['TempInstability'] = weather_data['HourlyDryBulbTemperature'].diff(3).abs().fillna(0)
weather_data['DryPeriodDuration'] = np.log1p(weather_data['HoursSinceRain'])

# Applying SMOTE
smote = SMOTE(random_state=42)
X_train_balanced, y_train_balanced = smote.fit_resample(X_train, y_train)
```

Feature Engineering guided by Meteorological Literature



```
# Add features based on article
wind_speed_threshold = df['HourlyWindSpeed'].quantile(0.75)
df['MonsoonWindSurge'] = (df['HourlyWindSpeed'].rolling(window=6, min_periods=1).max() >= wind_speed_threshold).astype(int)
df['PressureDropRate'] = df['HourlySeaLevelPressure'].diff(periods=6).fillna(0)
df['CumulativeHumidity'] = df['HourlyRelativeHumidity'].rolling(window=12, min_periods=1).mean().fillna(df['HourlyRelativeHumidity'])
df['TempInstability'] = df['HourlyDryBulbTemperature'].diff(periods=3).abs().fillna(0)
```

Model Implementation and Evaluation

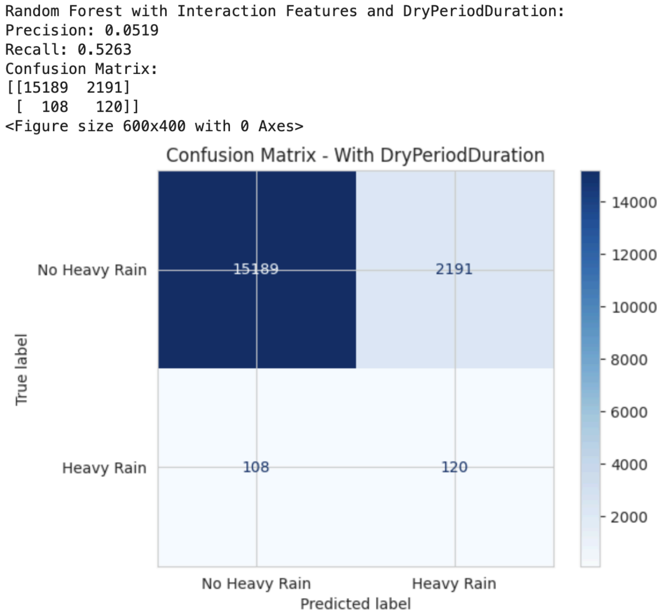
The Random Forest classifier emerged as the most feasible model for predicting cloudbursts, leveraging a comprehensive feature set that combined engineered atmospheric indicators and polynomial interaction terms. The model aimed at forecasting heavy rainfall occurrences within a three-hour lead time, defined specifically by precipitation exceeding 0.41 inches/hour.

Three distinct model configurations were systematically evaluated: one employing only base meteorological features, another adding polynomial interactions, and the final incorporating the DryPeriodDuration feature. The fully enhanced Random Forest model provided the best performance in predicting rare, extreme rainfall events.

- Evaluation on a temporal held-out test set produced the following metrics:
- Precision: 0.0519
- Recall: 0.5263

These metrics underscored the intentional optimization towards recall, prioritizing the identification of actual cloudburst occurrences even at the expense of increased false positives. Given the severe consequences of missed predictions in a disaster scenario, this tradeoff was strategically justified.

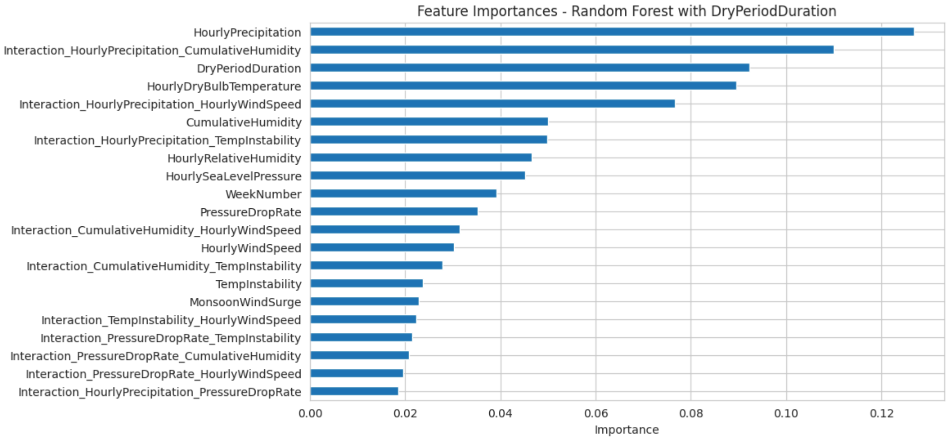
Best Performance Result with Confusion Matrix



```
Random Forest with Interaction Features and DryPeriodDuration:
Precision: 0.0519
Recall: 0.5263
Confusion Matrix:
[[15189  2191]
 [  108   120]]
<Figure size 600x400 with 0 Axes>
```

Feature importance analysis underscored the critical predictive roles of interaction features, notably the interaction between hourly precipitation and cumulative humidity, and base variables including hourly precipitation, temperature instability, and relative humidity. Importantly, adding DryPeriodDuration markedly improved model recall relative to earlier iterations (from 0.4474 base, 0.5702 interaction, to 0.5263 with dry period), demonstrating its relevance in capturing underlying atmospheric dynamics preceding heavy rainfall.

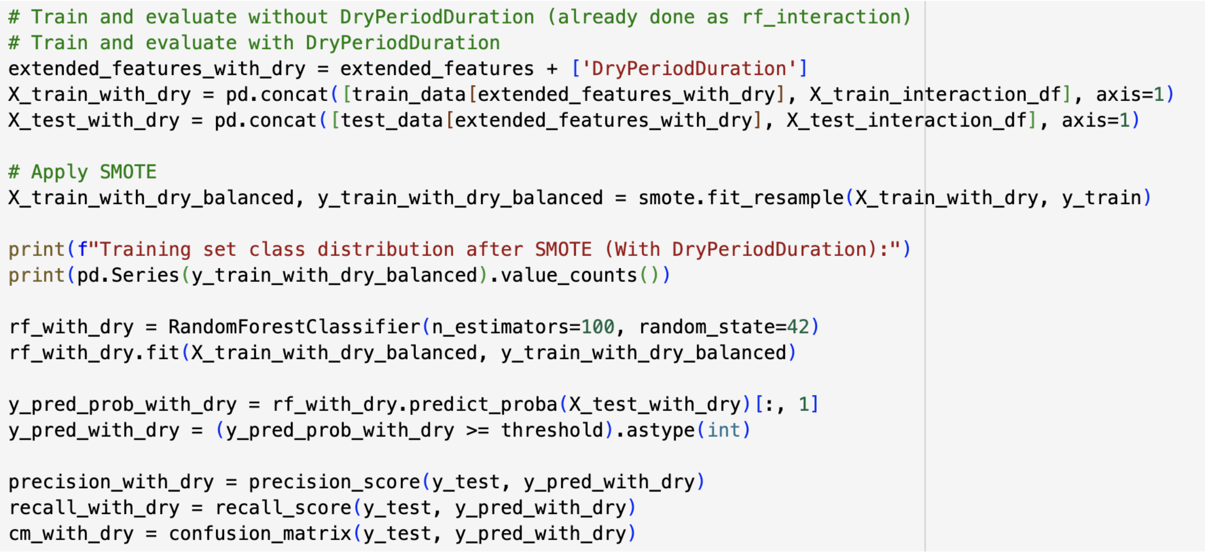
Best Model’s Feature Importance



The Random Forest model consistently outperformed earlier explored methodologies, including logistic regression and autoencoder-based anomaly detection, in terms of recall, robustness, and interpretability. Logistic regression exhibited limitations in modeling complex, nonlinear interactions inherent in meteorological data, while autoencoder methods struggled with stable threshold setting amidst observational noise and data irregularities.

The final Random Forest configuration balanced detection capability, interpretability, and resilience to data imperfections effectively. Future enhancements may explore further precision improvements through threshold optimization, ensemble techniques, and inclusion of additional meteorological datasets or satellite-derived features.

Random Forest Model Training Code



```
# Train and evaluate without DryPeriodDuration (already done as rf_interaction)
# Train and evaluate with DryPeriodDuration
extended_features_with_dry = extended_features + ['DryPeriodDuration']
X_train_with_dry = pd.concat([train_data[extended_features_with_dry], X_train_interaction_df], axis=1)
X_test_with_dry = pd.concat([test_data[extended_features_with_dry], X_test_interaction_df], axis=1)

# Apply SMOTE
X_train_with_dry_balanced, y_train_with_dry_balanced = smote.fit_resample(X_train_with_dry, y_train)

print(f"Training set class distribution after SMOTE (With DryPeriodDuration):")
print(pd.Series(y_train_with_dry_balanced).value_counts())

rf_with_dry = RandomForestClassifier(n_estimators=100, random_state=42)
rf_with_dry.fit(X_train_with_dry_balanced, y_train_with_dry_balanced)

y_pred_prob_with_dry = rf_with_dry.predict_proba(X_test_with_dry)[:, 1]
y_pred_with_dry = (y_pred_prob_with_dry >= threshold).astype(int)

precision_with_dry = precision_score(y_test, y_pred_with_dry)
recall_with_dry = recall_score(y_test, y_pred_with_dry)
cm_with_dry = confusion_matrix(y_test, y_pred_with_dry)
```

Ethical Considerations

The development and deployment of cloudburst prediction systems, particularly those utilizing machine learning (ML), carry significant ethical responsibilities. One of the most pressing concerns lies in the consequences of false positives versus false negatives. A false positive—incorrectly forecasting a cloudburst—may trigger unnecessary evacuations, economic losses, and public desensitization to future warnings. Over time, this could erode public trust in early warning systems. Conversely, false negatives—failing to detect an actual cloudburst—pose far graver risks, potentially resulting in property destruction, injury, or loss of life. In urban contexts with dense populations and vulnerable infrastructure, the cost of under-predicting extreme events is ethically unacceptable.

While most meteorological data is publicly sourced and anonymized, privacy and data security become relevant when integrating location-specific data streams from mobile apps, private sensors, or surveillance-enabled platforms. If real-time user mobility data is ever used to optimize evacuation models or emergency responses, strong safeguards must be in place to prevent misuse or unauthorized access.

Moreover, the application of ML introduces concerns around algorithmic transparency and fairness. Black-box models may yield accurate results but provide little insight into how decisions are made—a challenge in high-stakes, public-facing systems. Additionally, training models on biased or insufficient data can disproportionately affect marginalized communities by either overlooking their risk or exaggerating it.

To mitigate these concerns, prediction systems should incorporate model explainability features, prioritize false negative minimization, and include rigorous bias audits during training. Ethical review boards and cross-sector stakeholder consultations—especially involving urban planners and civil society—can further ensure that these systems are both effective and just.

AI Assistants Use and Process Reflection

AI tools—especially Grok, Claude, and ChatGPT—were foundational to every stage of my cloudburst prediction project. From defining the problem to refining model output, I used AI not just for implementation but as a dynamic collaborator that helped navigate ambiguity, automate repetitive tasks, and offer domain-specific insights.

In the early stage of data analysis, I began by exploring the distribution of cloudbursts in the dataset—specifically, identifying how many extreme rainfall events met the cloudburst threshold. This baseline check helped me understand the scale of the imbalance problem I would face later during modeling. I also generated correlation plots and exploratory visualizations to examine relationships between variables such as humidity, pressure, wind speed, and precipitation. AI assisted in guiding which visualization tools and methods were most appropriate, and helped interpret whether patterns suggested causal or merely correlative relationships. This process gave me a stronger foundation for meaningful feature engineering.

I broke the entire project into four stages: (1) analyze the data, (2) engineer features, (3) build models, and (4) apply refinements or adjustments. In each of these stages, I consulted Grok and Claude more frequently than ChatGPT, primarily due to their ease of use and contextual memory when writing or revising code. For instance, when implementing SMOTE to correct for class imbalance, or when creating engineered features like DryPeriodDuration or PressureDropRate, I frequently tested my logic by asking AI to validate assumptions, check syntax, and suggest additional transformations.

What I found most critical in working with AI tools was frequent and intentional interaction. Because generative AI is built on transformer-based architectures, each interaction acts as a token of influence—contributing to the “chain of thought” that shapes the final output. If I had only prompted the AI once and accepted its first response, the outcome would have diverged significantly from my evolving objectives. By breaking complex goals into smaller sub-tasks and offering continuous feedback—such as model performance metrics or revised research constraints—I was able to steer the AI toward results that were not just functional, but also contextually aligned with real-world forecasting needs.

Still, the use of AI had limitations. At times, suggestions made by Claude or Grok were theoretically sound but poorly suited for my data’s temporal structure or rare-event nature.

For instance, some thresholds recommended for classification didn't generalize well due to seasonal fluctuations in rainfall. This highlighted the importance of human oversight and iterative evaluation, especially when tuning hyperparameters or interpreting feature importances.

From this project, I gained not only technical skills in working with imbalanced classification and time-series forecasting, but also strategic intuition about how to structure collaboration with AI. I developed the ability to interrogate AI outputs, refine prompts, and merge domain knowledge with automated insights. These skills are transferable far beyond weather prediction—whether it's in finance, health, or other domains where understanding patterns in rare, high-stakes events is crucial.

Ultimately, my success in this novel technical domain was less about mastering every algorithm and more about mastering the workflow: decomposing problems, collaborating with AI iteratively, and validating each step with thoughtful skepticism.

Conclusion and Future Directions

Throughout this practicum, we developed and refined a machine learning (ML) pipeline aimed at accurately predicting rare and severe cloudburst events. Key findings from this project underscored the value of integrating domain-specific feature engineering—such as cumulative humidity, pressure drop rates, and dry period duration—to significantly improve predictive performance. Among evaluated methods, our Random Forest classifier, enhanced with polynomial interaction terms, consistently delivered superior recall and interpretability. Notably, our analysis highlighted that maintaining temporal integrity through weekly cluster sampling was critical to producing realistic performance estimates, effectively addressing concerns about temporal leakage commonly encountered in weather data modeling.

However, our ML approach also faced limitations. Primarily, class imbalance remained challenging, as cloudburst events represented only 0.49% of observations, complicating model training and evaluation. The modest precision scores obtained indicated persistent false-positive issues, necessitating careful threshold management. Additionally, the localized nature of cloudbursts meant that dataset granularity—both spatially and temporally—posed inherent constraints on predictive accuracy.

Moving forward, future improvements should include leveraging additional meteorological data sources such as satellite-derived precipitation measurements (e.g., NASA's IMERG), which could enrich the feature space and enhance spatial-temporal granularity. Exploring advanced neural network architectures, like Convolutional Neural Networks (CNNs) with Gramian Angular Fields (GAF), could further improve pattern recognition in complex meteorological data. Moreover, real-time ensemble modeling and adaptive threshold tuning might significantly reduce false positives, improving practical utility. Continued close collaboration with domain experts, rigorous ethical oversight, and transparent algorithmic explanations will ensure that the INDRA system provides robust, actionable insights for urban disaster preparedness.