Predicting and Forecasting Future Weather Impacts on Power Outages

Group

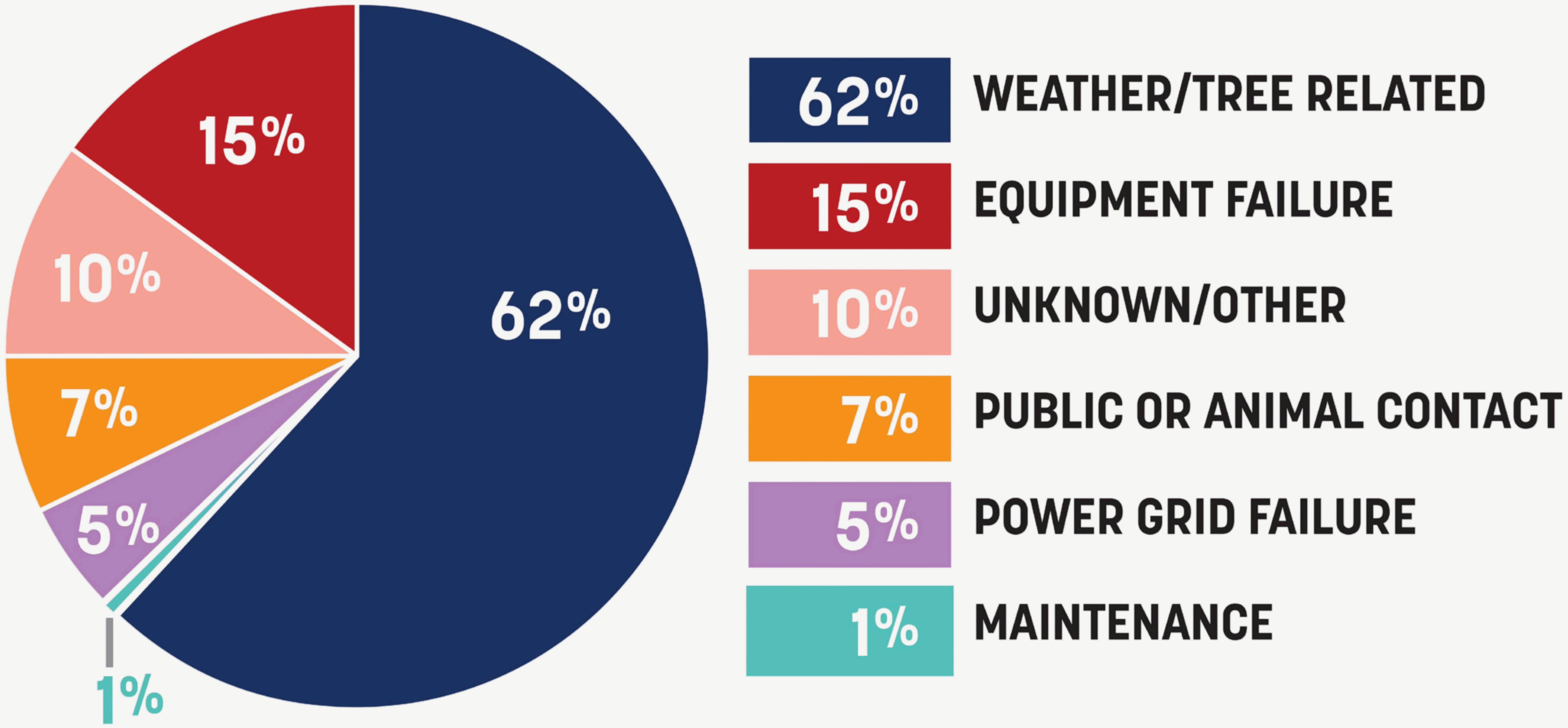
Akshat Gupta

Raghav Sarna

Shreyansh Singh

May 2025

MAJOR CAUSES OF POWER OUTAGES IN THE U.S.



Courtesy of Dr. Mladen Kezunovic

Why is it a High Impact Problem

Public Safety

Vulnerable populations (elderly, medical patients) rely on power for life-saving devices (e.g., ventilators).

[Ref: Greenwald et al., 2004, Apenteng et al., 2018]

Economic Stability

Businesses lose billions of dollars annually in the U.S. due to outages.

[Ref: Macmillan et al., 2023]

Emergency Responders

Emergency responders rely on electricity for communication and coordination.

[Ref: Adhikari et al., 2017]

Amongst many others............

Paper 1:

Negative Binomial Regression for Hurricane-Induced Power Outages

Table 2. Number of Data Points in Grid Cell Model by Company and Hurricane (Number of points in Zip Code model in parentheses)

	Company		
Hurricane	Duke Energy	Progress Energy	Total
Bonnie	—	14,723 (233)	14,723 (233)
Floyd	624 (85)	20,284 (302)	20,908 (387)
Fran	6,373 (14)	—	6,373 (14)
Total	6,997 (99)	35,007 (535)	42,004 (634)

Liu, Haibin & Davidson, Rachel & Rosowsky, David & Stedinger, Jery. (2005). Negative Binomial Regression of Electric Power Outages in Hurricanes. Journal of Infrastructure Systems - J INFRASTRUCT SYST. 11. 10.1061/(ASCE)1076-0342(2005)11:4(258).

Methodology

Used a Negative Binomial Regression model to analyze power outages caused by hurricanes

Observations

- Grid cell Based Model failed and Zip-Code based model was employed instead
- The model's applicability is limited as it includes company and hurricane indicator variables, restricting its use to the specific cases studied.

Performance

Pseudo-R^2 (Deviance-based): 78% (ZIp-Code)
48% (Grid Cell)

Limitations

- The model was developed using outage data from only three hurricanes.
- The study focuses on permanent faults and does not account for transient faults.
- The accuracy of the model is affected by missing details about factors like recent tree trimming, and other environmental variables.

Paper 2:

Predicting Storm Outages Through New Representations of Weather and Vegetation

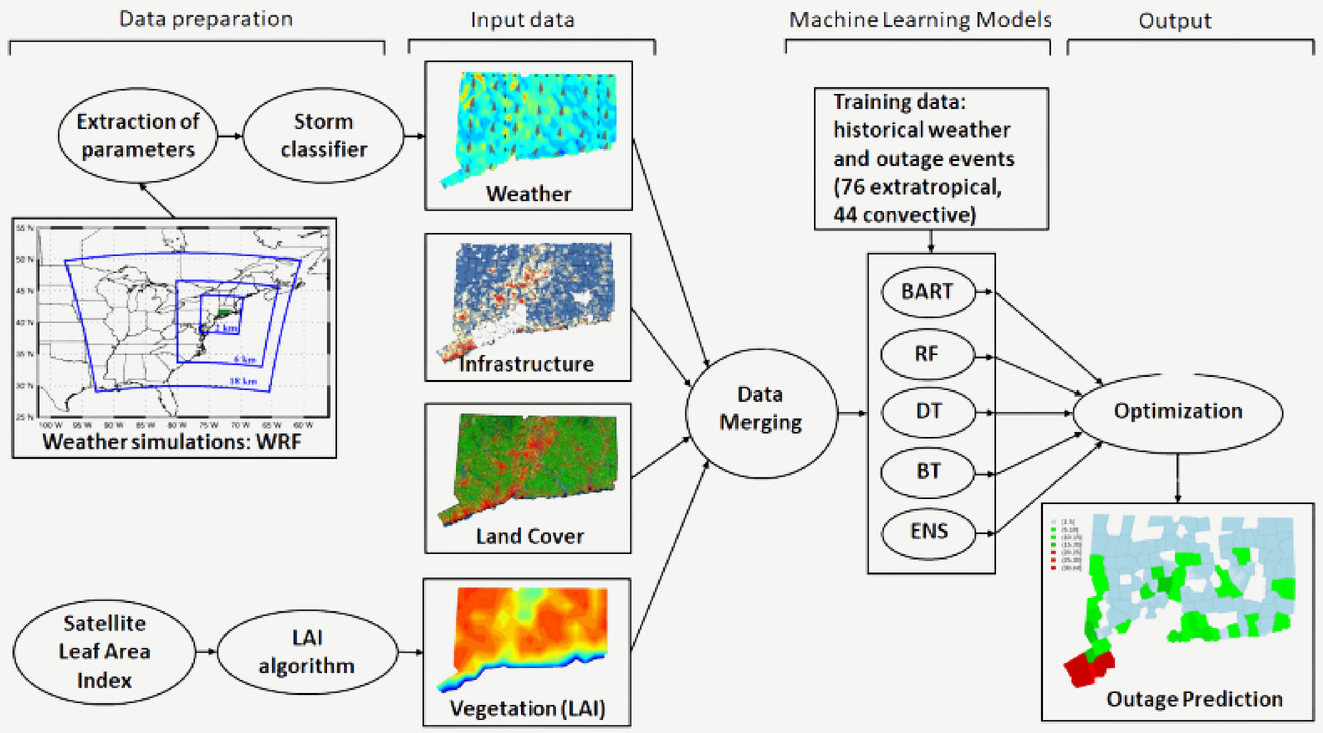


FIGURE 1. **The Outage Prediction Model architecture.**

median absolute percentage error (MdAPE)	*mean absolute percentage error* (MAPE)	*Nash-Sutcliffe Efficiency* (NSE
MdAPE	**MAPE**	**NSE**
43% (+9%)	**59%** (+9%)	**0.53** (+13%)

D. Cerrai et al., "Predicting Storm Outages Through New Representations of Weather and Vegetation," in IEEE Access, vol. 7, pp. 29639-29654, 2019, doi: 10.1109/ACCESS.2019.2902558.

Methodology

- High-res WRF forecasts, satellite LAI, land cover, and utility data on a unified 2-km grid.
- Multiple models (DT, GBM, RF, ENS, BART) with PCA for dimensionality reduction.

Observations

- Fills gaps using AR(1) and Gaussian filtering.
- Differentiates storm types for tailored variable selection.
- Captures complex interactions among weather, vegetation, and infrastructure.
- 24-hour forecasts align strongly with historical data.

Limitations

Using a fixed 2-km grid might not capture localized variations in weather and vegetation properly.

Paper 3:

Predicting Weather-related Power Outages in Distribution Grid

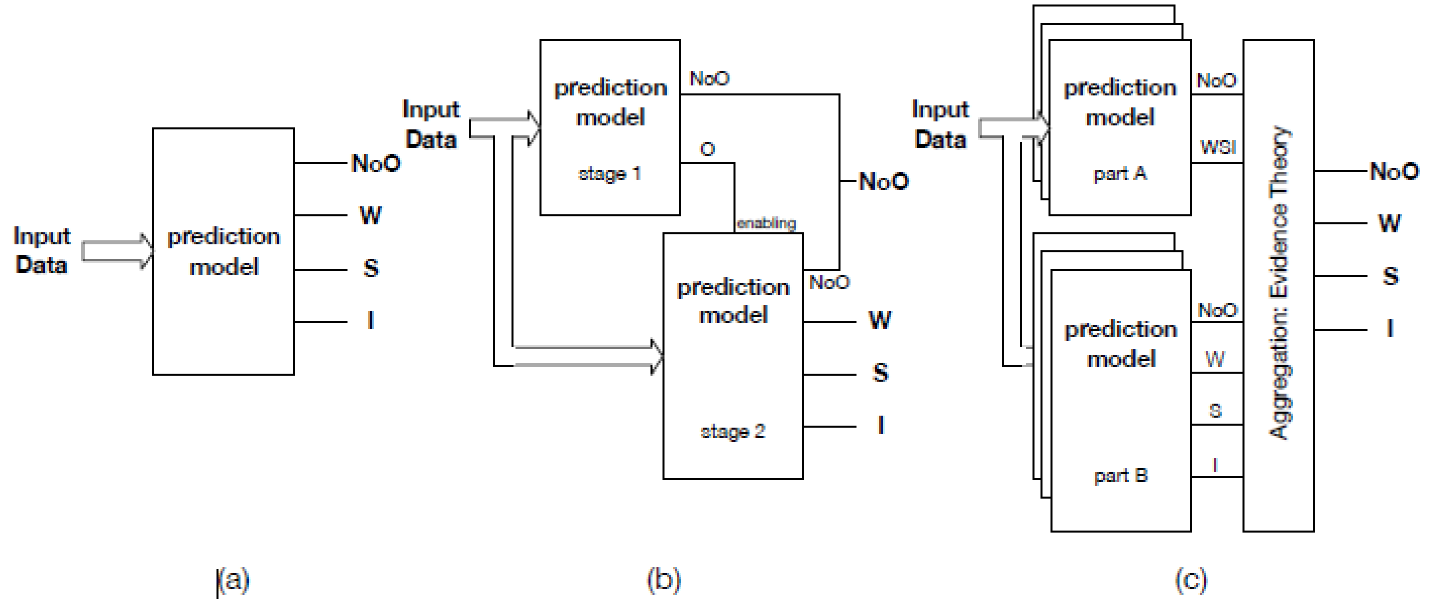


Figure 1. System Architecture

Y. Kor, M. Z. Reformat and P. Musilek, "Predicting Weather-related Power Outages in Distribution Grid," 2020 IEEE Power & Energy Society General Meeting (PESGM), Montreal, QC, Canada, 2020, pp. 1-5, doi: 10.1109/PESGM41954.2020.9281829.

Methodology

Developed a WoutPS (Weather Outage Prediction System) using and Ensemble Model with multiple prediction models

Observations

- Ensemble Model worked way better than the first 2 architectures
- Dempster-Shafer theory (DST) was used for Reasoning and used to find Pignistic Probabilities

Performance

Table II
PRECISION, RECALL, AND F_1 SCORE REPORT ON TEST DATA FOR ARCHITECTURES a, b, c

Metric	NoO	Wind	Snow	Icing	Macro Ave.
Precision a	0.93	0.65	0.83	0.73	0.79
Recall a	0.96	0.41	0.68	**0.76**	0.70
F_1 score a	0.94	0.50	0.75	0.74	0.74
Precision b	0.93	0.66	0.85	0.74	0.80
Recall b	0.96	0.41	0.68	0.73	0.70
F_1 score b	0.95	0.51	0.76	0.74	0.74
Precision c	0.93	**0.68**	**0.86**	**0.81**	**0.82**
Recall c	**0.97**	**0.42**	**0.73**	0.73	**0.71**
F_1 score c	**0.95**	**0.52**	**0.79**	**0.77**	**0.76**

Table III
PIGNISTIC PROBABILITY FOR VARIOUS SAMPLES AND THE REFERENCE VALUES

Id	Reference	NoO	Wind	Snow	Icing
1	NoO	**0.9981**	0.0005	0.0008	0.0006
2	NoO	0.0579	**0.3232**	**0.2362**	**0.3828**
3	Wind	0.2342	**0.7181**	0.0246	0.0230
4	Wind	**0.8203**	0.1684	0.0056	0.0056
5	Snow	0.0286	0.0145	**0.9402**	0.0167
6	Snow	**0.6642**	0.0798	0.1836	0.0724
7	Icing	0.0297	0.0339	0.0559	**0.8804**
8	Icing	0.0058	**0.6912**	0.1294	0.1736

Limitations

Pignistic Probability show large differences for different Classifiers resulting in less trust

Gap

(BART) (Table 4, first box). The typical feature of some of these models was a remarkable overestimation of low impact events, as shown in W15 and H17, which led to very high values of MAPE.

Paper 2

Despite the overall improved forecast skill of the conditioned OPM, QWD misclassified some moderate and low-severity events

F. Yang, P. Watson, M. Koukoula and E. N. Anagnostou, "Enhancing Weather-Related Power Outage Prediction by Event Severity Classification,"

respectively. The classification F1 Scores for low-, moderate-, and high-severity classes using analysis weather data were 0.62, 0.62, and 1, respectively, while using forecast weather data, they were 0.57, 0.67, and 1, respectively.

- The existing Models have demonstrated reasonably good forecasting with the outage prediction model (OPM).
- But they still *struggle with misclassification of moderate and low-severity storms.*
- **Our project has addressed this gap by refining storm classification methodologies to improve classification for moderate and low-severity storms as well.**
- Even lower/modern-severity storms can cause outages, affecting communities, businesses, and emergency services.

EAGLE-I Dataset

The dataset consists of county-level electricity outage records in the United States from 2014 to 2023. It provides outage estimates at 15-minute intervals, covering 92% of electricity customers across all 50 U.S. states by 2023. It makes it the most comprehensive publicly available record of U.S. electricity outages. The data was compiled as part of the EAGLE-I (Environment for Analysis of Geo-Located Energy Information) project, a GIS-based platform developed by Oak Ridge National Laboratory (ORNL).

How they collected:

- Web scraping of publicly available outage reports from over 456 U.S. electrical utilities.
- Automated parsers extracting data from utility websites every 15 minutes.

Privacy and Security:

- The dataset only includes aggregated county-level data, not individual household or customer details.

Data Accuracy and Bias:

- Not all utilities report outages, so coverage is not 100%
- A Data Quality Index (DQI) was developed by them to assess coverage completeness and accuracy across FEMA regions.

Time Period:

10 Years (2014-2023)

Coverage:

3044 of 3222 Counties

Granularity:

15 Minute Intervals

NOAA Dataset

This Dataset consists records of all Storm related events and has the following columns :

BEGIN_YEARMONTH
BEGIN_DAY
BEGIN_TIME
END_YEARMONTH
END_DAY
END_TIME
EPISODE_ID
EVENT_ID
STATE
STATE_FIPS
YEAR
MONTH_NAME
EVENT_TYPE
CZ_TYPE
CZ_FIPS
CZ_NAME
WFO
BEGIN_DATE_TIME
CZ_TIMEZONE
END_DATE_TIME

INJURIES_DIRECT
INJURIES_INDIRECT
DEATHS_DIRECT
DEATHS_INDIRECT
DAMAGE_PROPERTY
DAMAGE_CROPS
SOURCE
MAGNITUDE
MAGNITUDE_TYPE
FLOOD_CAUSE
CATEGORY
TOR_F_SCALE
TOR_LENGTH
TOR_WIDTH
TOR_OTHER_WFO
TOR_OTHER_CZ_STATE
TOR_OTHER_CZ_FIPS
TOR_OTHER_CZ_NAME
BEGIN_RANGE
BEGIN_AZIMUTH
BEGIN_LOCATION

END_RANGE
END_AZIMUTH
END_LOCATION
BEGIN_LAT
BEGIN_LON
END_LAT
END_LON
EPISODE_NARRATIVE
EVENT_NARRATIVE
DATA_SOURCE

Documents

1. The occurrence of storms and other significant weather phenomena having sufficient intensity to cause loss of life, injuries, significant property damage, and/or disruption to commerce;
2. Rare, unusual, weather phenomena that generate media attention, such as snow flurries in South Florida or the San Diego coastal area
3. Other significant meteorological events, such as record maximum or minimum temperatures or precipitation that occur in connection with another event.

Time Period:

74 Years (1950-2023)

We are using only 10 years
(2014-2023)

PRISM Climate Dataset

The PRISM dataset provides high-resolution, county-level climate data across the United States, developed by the PRISM Climate Group at Oregon State University. Data that we are using includes daily estimates of key meteorological variables such as minimum temperature (tmin), maximum temperature (tmax), average temperature (tavg), precipitation (ppt), and stability indices. Covering the period from 1981 to the present, the dataset integrates ground-based observations, satellite data, and topographic information using sophisticated interpolation techniques.

How they collected:

- Gridded climate data generated using advanced spatial climate modeling techniques.
- Daily values are available at a ~4 km resolution and aggregated to the county level.

Privacy and Security:

The dataset contains only aggregated geospatial and meteorological data, no personal, household, or identifiable customer data is included.

Data Accuracy and Bias:

- Uses statistical models to correct for topographic and geographic biases in climate data
- Applies rigorous quality control and weighting of weather station data to reduce errors and ensure reliable interpolation

Time Period:

43 Years (1981-2023)

We are using only 10 years (2014-2023)

Coverage:

All Counties

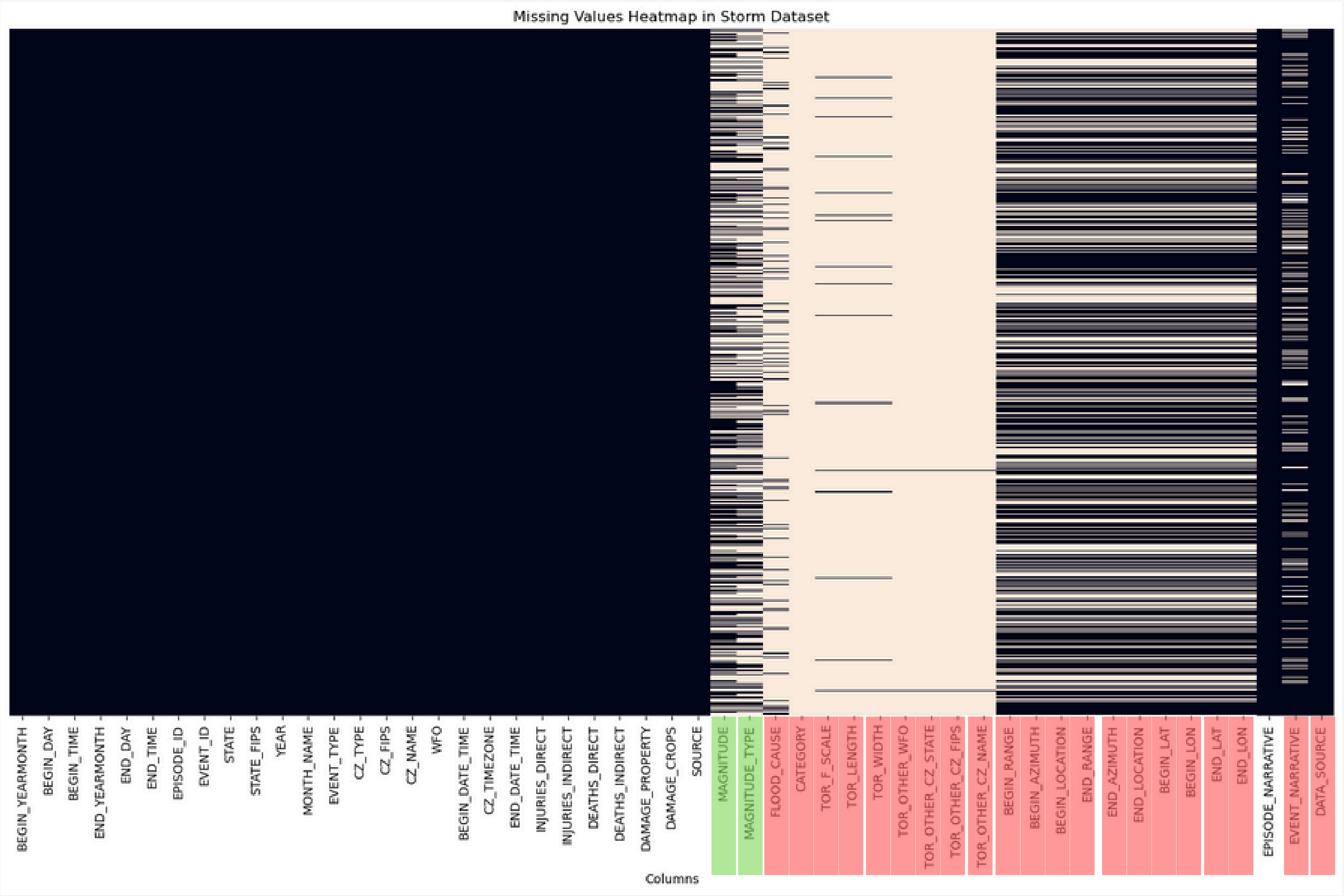
Granularity:

Daily Data

Feature Preprocessing & Selection

1.Cleaned up the data:

- Removing redundant features
- One hot encoding
- Fixing date time
- Converting to same time zone
- Fixing data types
- (...and more)



Red = Removed
Green = Imputed

2. **Added a new feature of duration (by subtracting end and start time)**

3. **Merging the three datasets based on location and time**

Feature Preprocessing & Selection

Magnitude Imputation

1. A RandomForestRegressor was trained on relevant features to predict missing 'MAGNITUDE' values.
2. Then, a RandomForestClassifier used the imputed or existing magnitudes to infer missing 'MAGNITUDE_TYPE'.
3. The imputed values were added back to the original dataset for further analysis.

NLP Integration

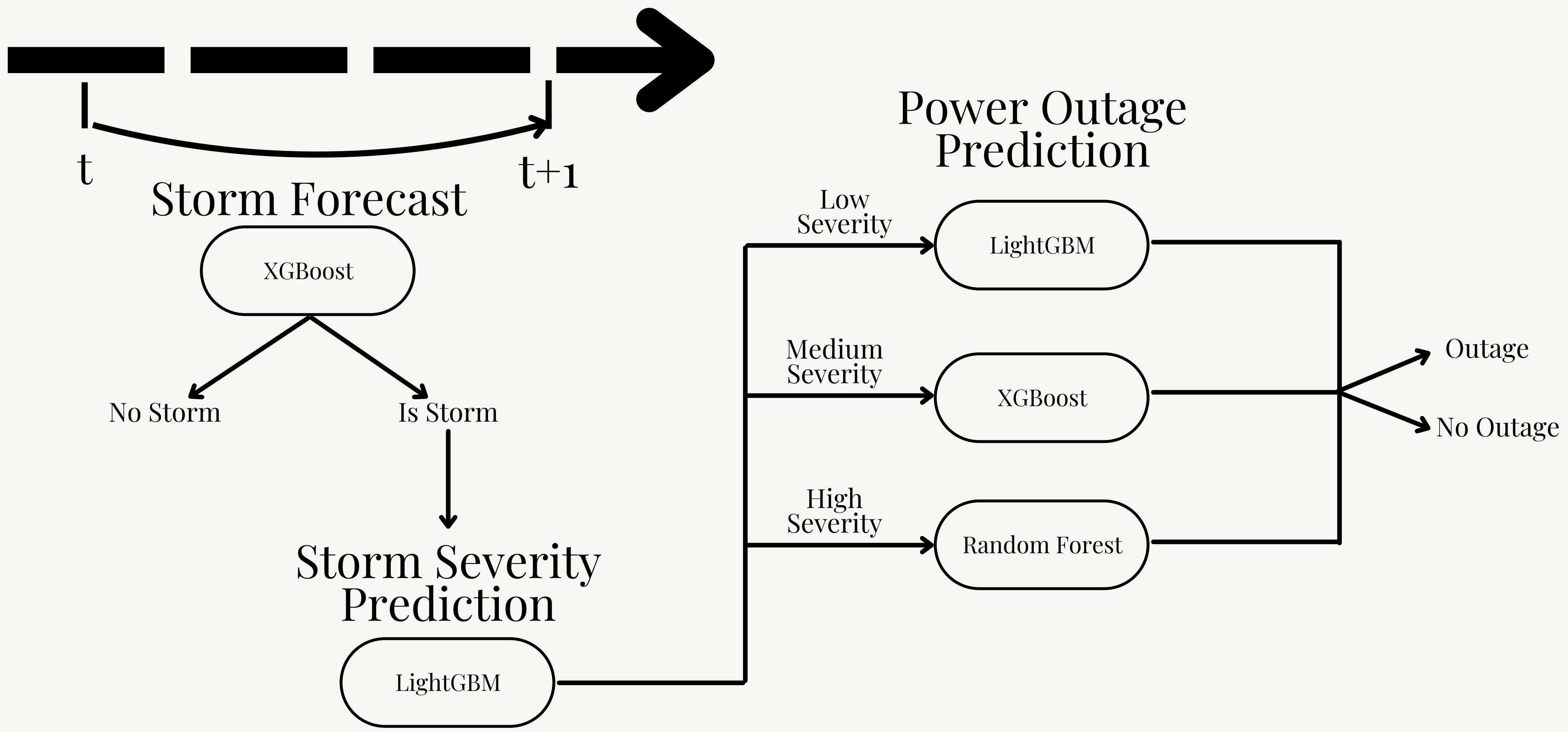
1. We preprocessed the description column by converting text to lowercase and removing non-alphabetic characters.
2. We tokenized the cleaned text and removed common English stopwords.
3. Lemmatization was applied to reduce words to their base form.
4. The cleaned tokens were joined back into a string and saved as desc_clean.

1-Hour Time Lag

1. The dataset is lagged by 1 hour to capture immediate precursors of storm events.
2. For each storm, it selects the nearest climate record occurring after a 1-hour delay.

Overview of ML Methodology

- **Objective:**
  - Predict storm occurrence, severity, and power outages using a multi-stage ML pipeline.
- **Pipeline Structure:**
  - Stage 1: Identify storm events (binary classification).
  - Stage 2: Classify storm severity (multi-class).
  - Stage 3: Predict power outages for each severity level (binary classification).



- **Key Features:**
  - Sequential processing of weather, damage, and text-derived features.
  - Modular design for scalability and real-time predictions.
- **Why This Approach?:**
  - Handles complex, non-linear relationships in weather and outage data.
  - Supports diverse tasks (binary and multi-class) with high accuracy.

Storm Forecasting

Random Forest

```
rf_model = RandomForestClassifier(
    n_estimators=201,
    max_depth=38,
    min_samples_split=5,
    min_samples_leaf=2,
    random_state=42
)
rf_model.fit(X_train_scaled, y_train)
```

→

```
Storm Prediction Test Metrics:
Accuracy : 0.9218
Precision: 0.9292
Recall   : 0.9141
F1 Score : 0.9216
```

XGBoost

```
xgb_model = xgb.XGBClassifier(
    n_estimators=271,
    max_depth=27,
    min_child_weight=1,
    subsample=0.9496561162982775,
    colsample_bytree=0.6753754188169535,
    learning_rate=0.13297767767696025,
    random_state=42,
    eval_metric='logloss',
    use_label_encoder=False
)
xgb_model.fit(X_train_full_scaled, y_train_full)
```

→

```
Storm Prediction Test Metrics
Accuracy : 0.9303
Precision: 0.9344
Recall   : 0.9264
F1 Score : 0.9304
```

MLP

```
mlp_model = MLPClassifier(
    hidden_layer_sizes=(100, 100),
    learning_rate_init=0.0014038319709889086,
    alpha=0.00011678125845319615,
    solver='adam',
    max_iter=1000,
    random_state=42,
    early_stopping=True,
    validation_fraction=0.1
)
mlp_model.fit(X_train_full_scaled, y_train_full)
```

→

```
Storm Prediction Test Metrics
Accuracy : 0.8963
Precision: 0.9069
Recall   : 0.8846
F1 Score : 0.8956
```

CatBoost

```
# Train CatBoost model with best hyperparameters directly
cat_model = cb.CatBoostClassifier(
    depth=10,
    learning_rate=0.2607409030513983,
    iterations=300,
    l2_leaf_reg=4.939014260456282,
    border_count=70,
    random_seed=42,
    task_type='CPU',
    verbose=0
)
cat_model.fit(X_train_full_scaled, y_train_full)
```

→

```
Storm Prediction Test Metrics
Accuracy : 0.9104
Precision: 0.9192
Recall   : 0.9010
F1 Score : 0.9100
```

LightGBM

```
lgb_model = lgb.LGBMClassifier(
    num_leaves=95,
    max_depth=17,
    learning_rate=0.27723713319564375,
    n_estimators=300,
    min_child_samples=21,
    subsample=0.882075261917957,
    colsample_bytree=0.6619967560096401,
    random_state=42,
    objective='binary',
    metric='binary_logloss'
)
```

→

```
Storm Prediction Test Metrics
Accuracy : 0.9283
Precision: 0.9322
Recall   : 0.9246
F1 Score : 0.9284
```

FNN

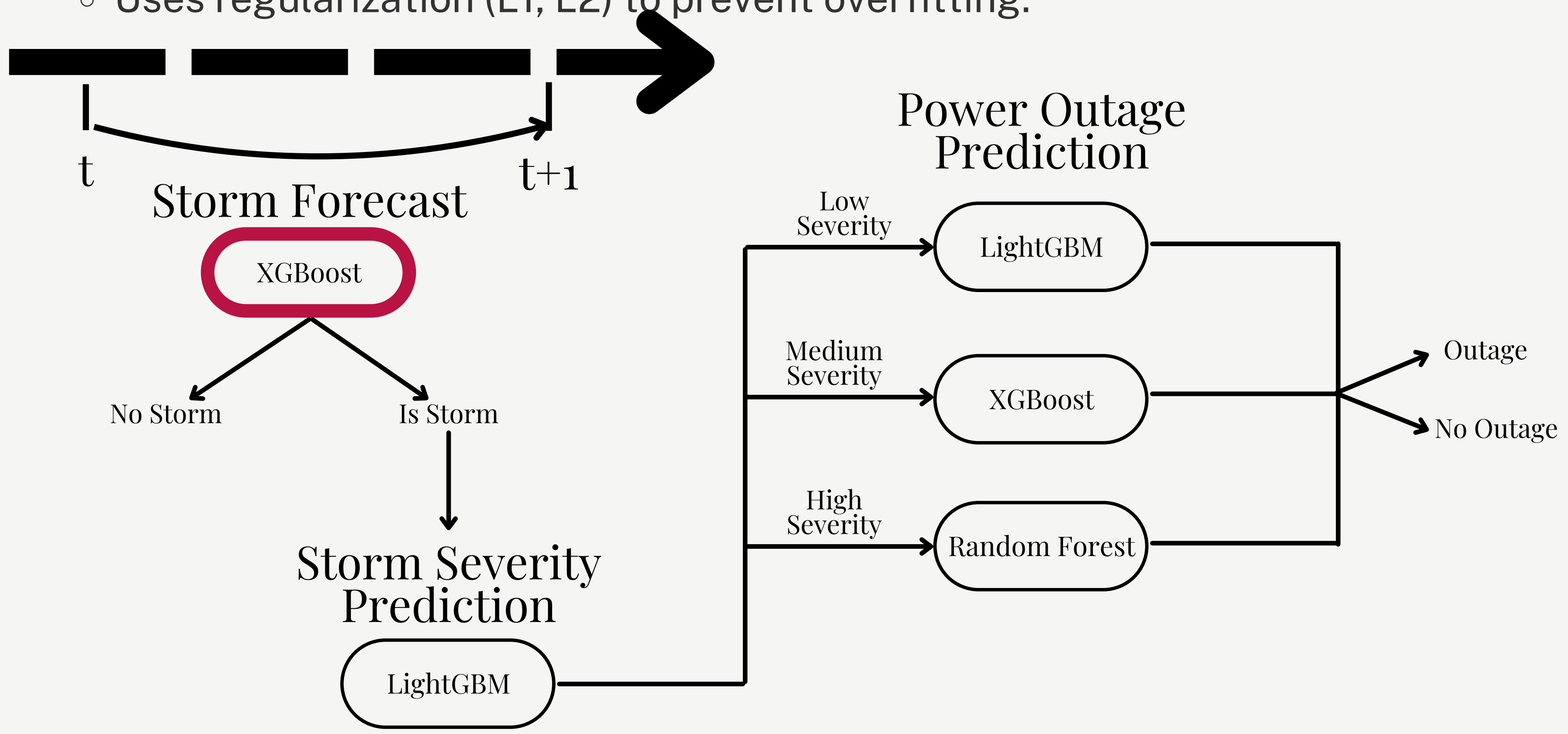
```
best_params = {
    'n_layers': 3,
    'n_units_l0': 234,
    'dropout_l0': 0.15187251202817453,
    'n_units_l1': 184,
    'dropout_l1': 0.2551136103189102,
    'n_units_l2': 241,
    'dropout_l2': 0.10855025262124199,
    'learning_rate': 0.0007371892834085776,
    'batch_size': 64
}
```

→

```
Test Accuracy:  0.8975
Test F1 Score:  0.8962
Test Precision: 0.9127
Test Recall:    0.8803
```

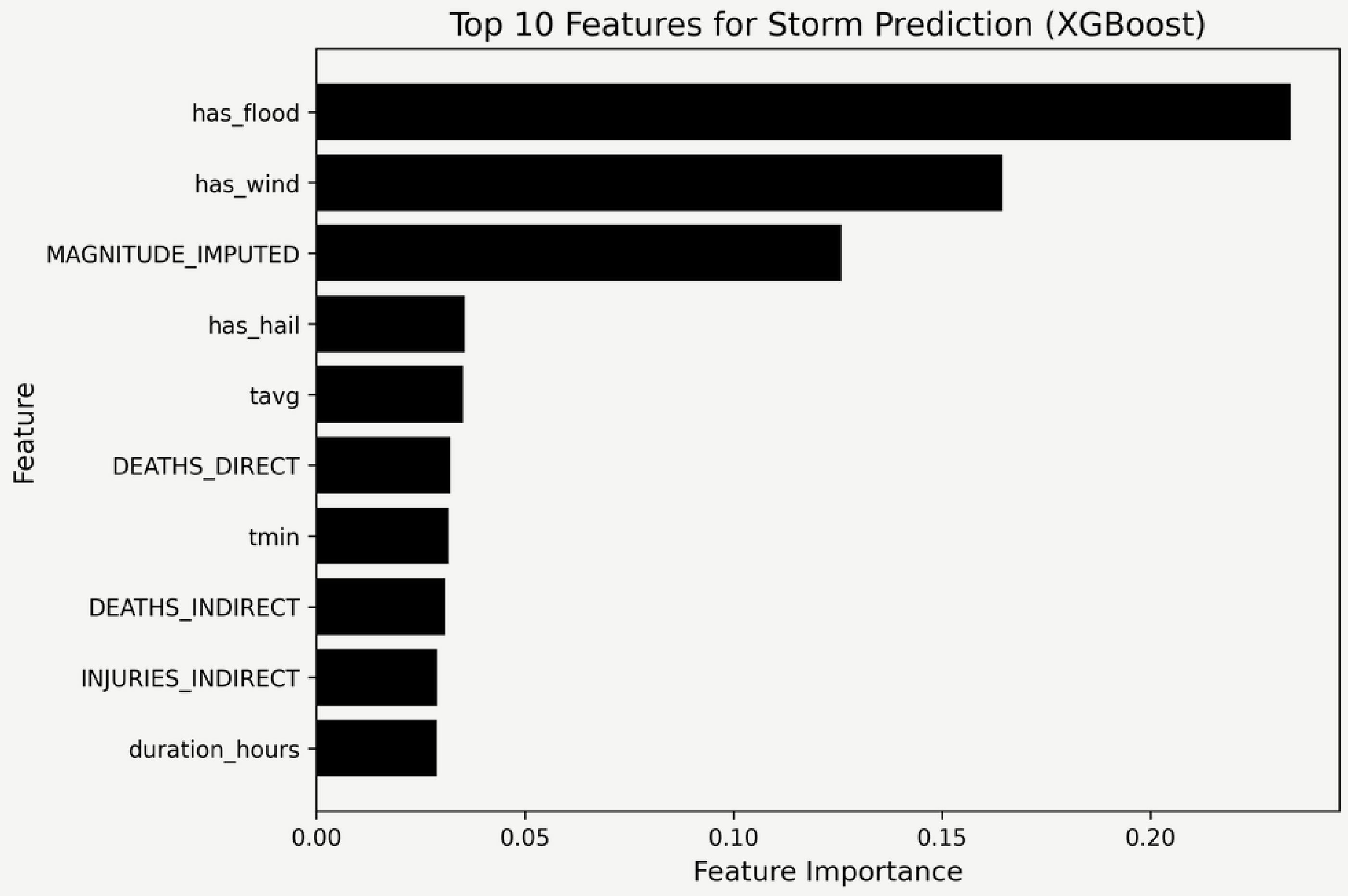
Storm Forecasting - XGBoost

- **Task:**
  - Binary classification (is_storm_lagged: 0=non-storm, 1=storm).
- **Model:**
  - XGBoost (n_estimators=271, max_depth=27, min_child_weight=1, subsample=0.9496561162982775, colsample_bytree=0.6753754188169535, learning_rate=0.13297767767696025, random_state=42, eval_metric='logloss', use_label_encoder=False).
- **Why XGBoost?:**
  - Excels in structured data with gradient boosting.
  - Handles high-dimensional data (21 features) with robustness.
  - Fast and scalable for large datasets.
- **How It Works:**
  - Iteratively builds trees, minimizing a loss function (log-loss for binary).
  - Uses regularization (L1, L2) to prevent overfitting.



Classification Report:

	precision	recall	f1-score	support
0	0.92	0.93	0.93	20493
1	0.93	0.93	0.93	21562
accuracy			0.93	42055
macro avg	0.93	0.93	0.93	42055
weighted avg	0.93	0.93	0.93	42055



Severity Prediction

Random Forest

```
rf_model = RandomForestClassifier(
    n_estimators=201,
    max_depth=38,
    min_samples_split=5,
    min_samples_leaf=2,
    random_state=42
)
rf_model.fit(X_train_scaled, y_train)
```

→

```
Severity Prediction Test Metrics:
Accuracy : 0.9033
Precision: 0.9033
Recall   : 0.9033
F1 Score : 0.9033
```

XGBoost

```
xgb_model = XGBClassifier(
    n_estimators=403,
    max_depth=10,
    learning_rate=0.09065400280278058,
    subsample=0.933968095670629,
    colsample_bytree=0.5647574078202744,
    gamma=0.00017586655077512627,
    min_child_weight=2,
    use_label_encoder=False,
    eval_metric='logloss',
    random_state=42
)
```

→

```
Severity Prediction Test
Accuracy : 0.9157
Precision: 0.9158
Recall   : 0.9157
F1 Score : 0.9157
```

LightGBM

```
final_params = {
    'objective': 'multiclass',
    'num_class': len(y_train.unique()),
    'metric': 'multi_logloss',
    'boosting_type': 'gbdt',
    'verbose': -1,
    'random_state': 42,
    'n_estimators': 943,
    'max_depth': 14,
    'learning_rate': 0.299798636793099,
    'subsample': 0.6815843772645557,
    'colsample_bytree': 0.967548003596196,
    'min_child_weight': 1.2892385914567714,
    'reg_alpha': 0.009927790289736168,
    'reg_lambda': 1.7396121269192825e-06
}
```

→

```
Severity Prediction Test Metrics
Accuracy : 0.9308
Precision: 0.9309
Recall   : 0.9308
F1 Score : 0.9308
```

CatBoost

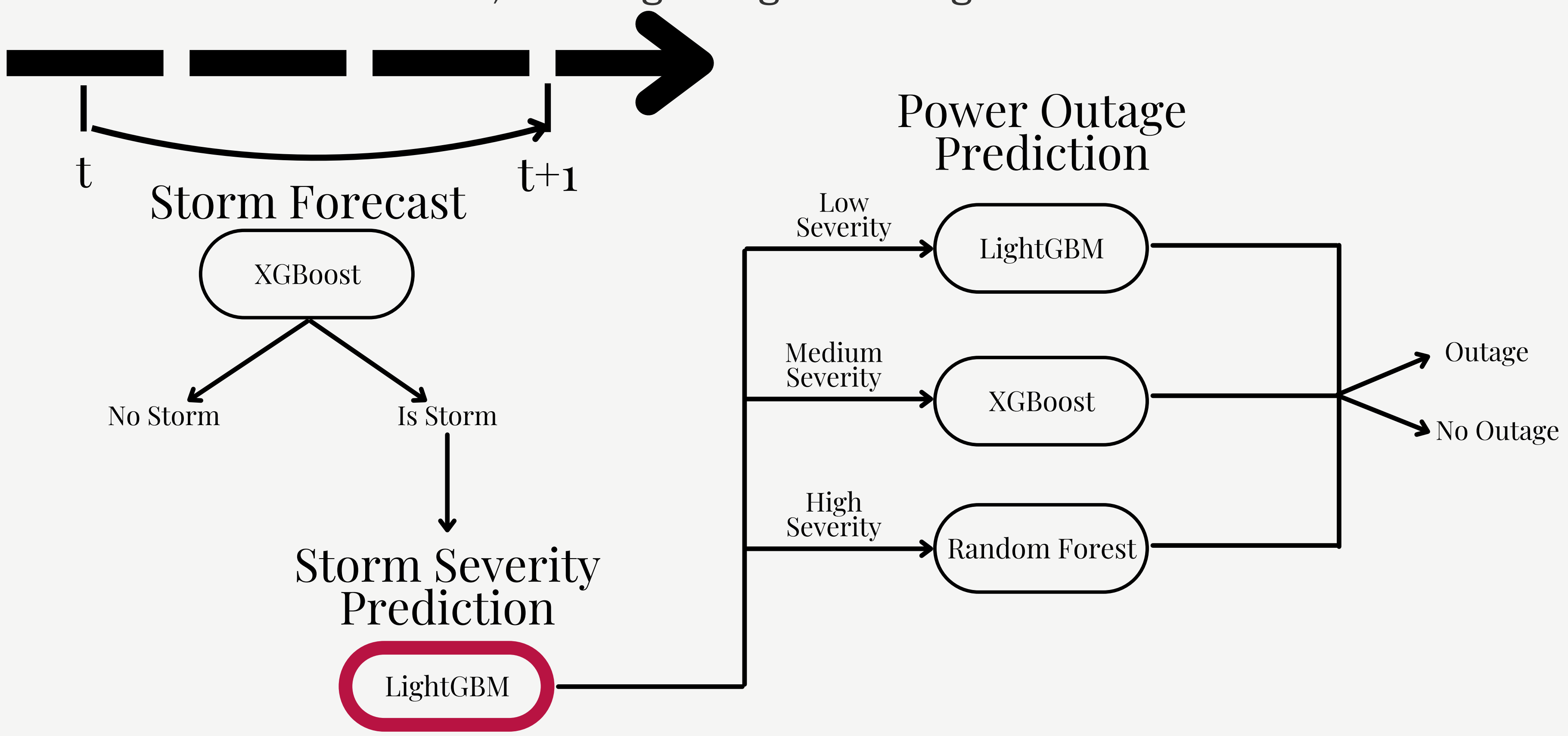
```
# Best parameters from Optuna
best_params = {
    'iterations': 771,
    'depth': 10,
    'learning_rate': 0.1358607927072531,
    'l2_leaf_reg': 3.698596513420882e-08,
    'bagging_temperature': 0.08933994426599372,
    'random_strength': 0.10544889935311012,
    'border_count': 178,
    'loss_function': 'MultiClass',
    'random_seed': 42,
    'verbose': 0
}
cat_model = CatBoostClassifier(**best_params)
cat_model.fit(X_train_scaled, y_train)
```

→

```
Severity Prediction Test Metrics
Accuracy : 0.9094
Precision: 0.9096
Recall   : 0.9094
F1 Score : 0.9095
```

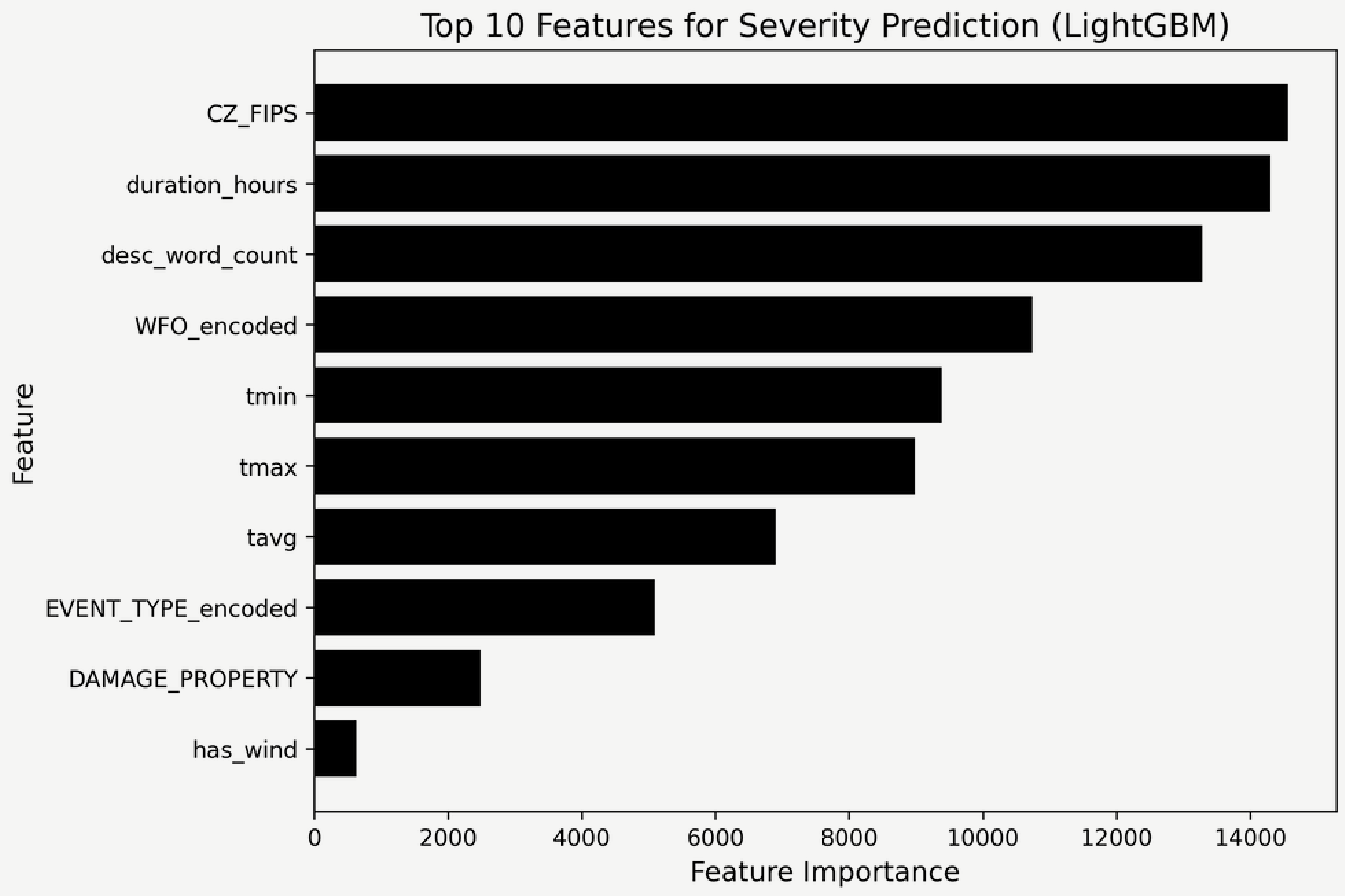
Severity Prediction - LightGBM

- **Task:**
  - Multi-class classification (severity_class: 0=low, 1=medium, 2=high).
- **Model:**
  - LightGBM (objective='multiclass', num_class=len(y_train.unique()), metric='multi_logloss', boosting_type='gbdt', verbose=-1, random_state=42, n_estimators=943, max_depth=14, learning_rate=0.299798636793099, subsample=0.6815843772645557, colsample_bytree=0.967548003596196, min_child_weight=1.2892385914567714, reg_alpha=0.009927790289736168, reg_lambda=1.7396121269192825e-06).
- **Why LightGBM?:**
  - Faster training than traditional gradient boosting.
  - Optimized for large datasets with encoded categorical features (e.g., EVENT_TYPE_encoded).
- **How It Works:**
  - Uses histogram-based learning for efficiency.
  - Builds trees leaf-wise, focusing on high-error regions.



Classification Report:

	precision	recall	f1-score	support
0	0.97	0.96	0.97	6674
1	0.90	0.90	0.90	6320
2	0.91	0.92	0.91	6614
10	1.00	1.00	1.00	1787
accuracy			0.93	21395
macro avg	0.94	0.94	0.94	21395
weighted avg	0.93	0.93	0.93	21395



Low Severity

Random Forest

```
rf_model = RandomForestClassifier(
    n_estimators=262,
    max_depth=41,
    min_samples_split=3,
    min_samples_leaf=1,
    max_features='log2',
    bootstrap=False,
    random_state=42,
    n_jobs=-1
)
```

```
Accuracy: 0.9774
Precision (weighted): 0.9774
Recall (weighted): 0.9774
F1 Score (weighted): 0.9774
```

XGBoost

```
xgb_model = XGBClassifier(
    n_estimators=186,
    max_depth=15,
    learning_rate=0.11082277907124656,
    subsample=0.881744658009053,
    colsample_bytree=0.9175879642068498,
    gamma=1.6112766577235877,
    min_child_weight=3,
    use_label_encoder=False,
    eval_metric='logloss',
    random_state=42
)
```

```
Accuracy: 0.9763
Precision (weighted): 0.9767
Recall (weighted): 0.9763
F1 Score (weighted): 0.9764
```

LightGBM

```
lgb_model = lgb.LGBMClassifier(
    n_estimators = 319,
    learning_rate = 0.047847333909262976,
    num_leaves = 281,
    max_depth = 20,
    min_child_samples = 22,
    subsample = 0.527136639688917,
    colsample_bytree = 0.8326768083509417,
    random_state = 42
)
lgb_model.fit(X_train_scaled, y_train)
```

```
Accuracy: 0.9783
Precision (weighted): 0.9784
Recall (weighted): 0.9783
F1 Score (weighted): 0.9783
```

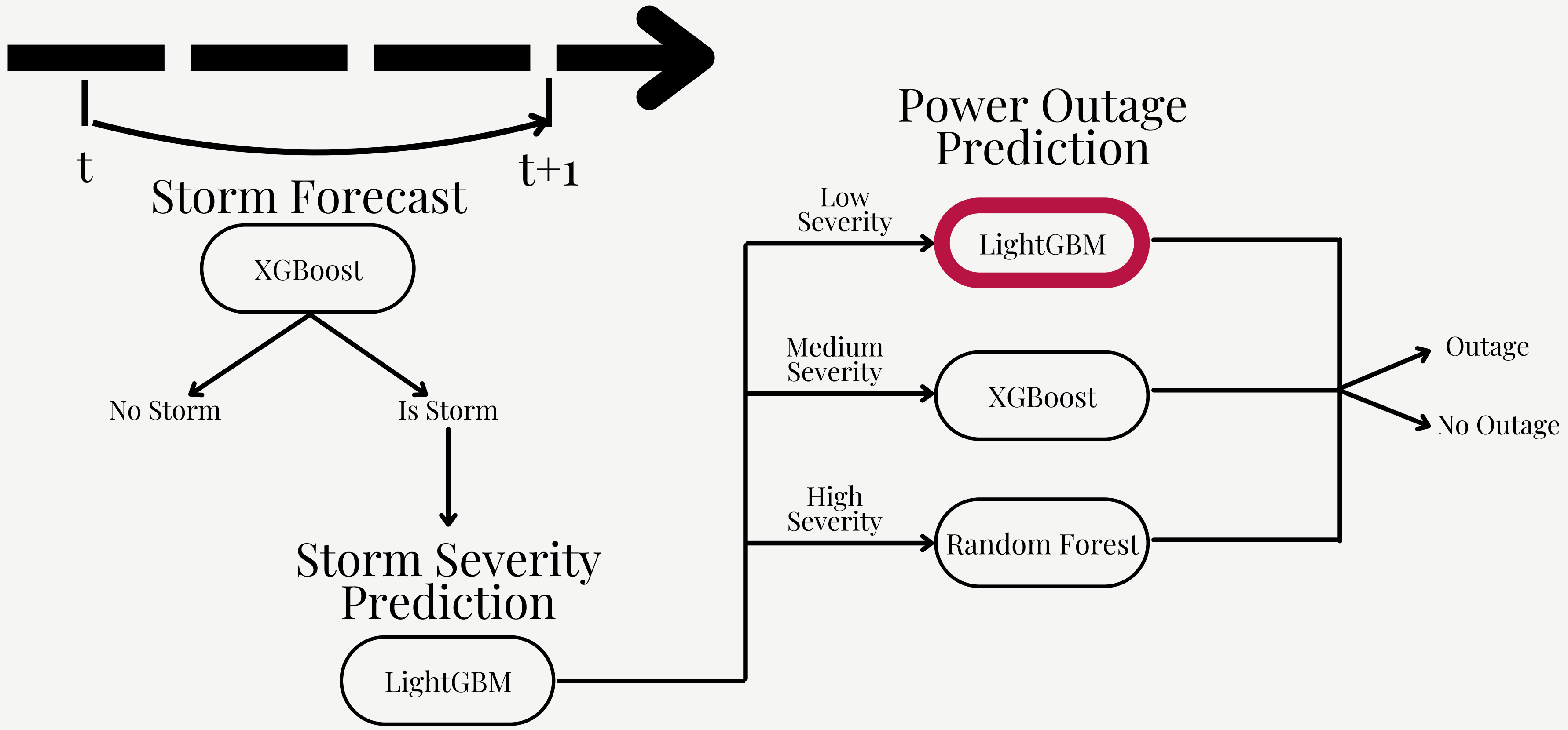
CatBoost

```
# Best parameters from Optuna
best_params = {
    'iterations': 771,
    'depth': 10,
    'learning_rate': 0.1358607927072531,
    'l2_leaf_reg': 3.698596513420882e-08,
    'bagging_temperature': 0.08933994426599372,
    'random_strength': 0.10544889935311012,
    'border_count': 178,
    'loss_function': 'MultiClass',
    'random_seed': 42,
    'verbose': 0
}
cat_model = CatBoostClassifier(**best_params)
cat_model.fit(X_train_scaled, y_train)
```

```
Accuracy: 0.9778
Precision (weighted): 0.9780
Recall (weighted): 0.9778
F1 Score (weighted): 0.9778
```

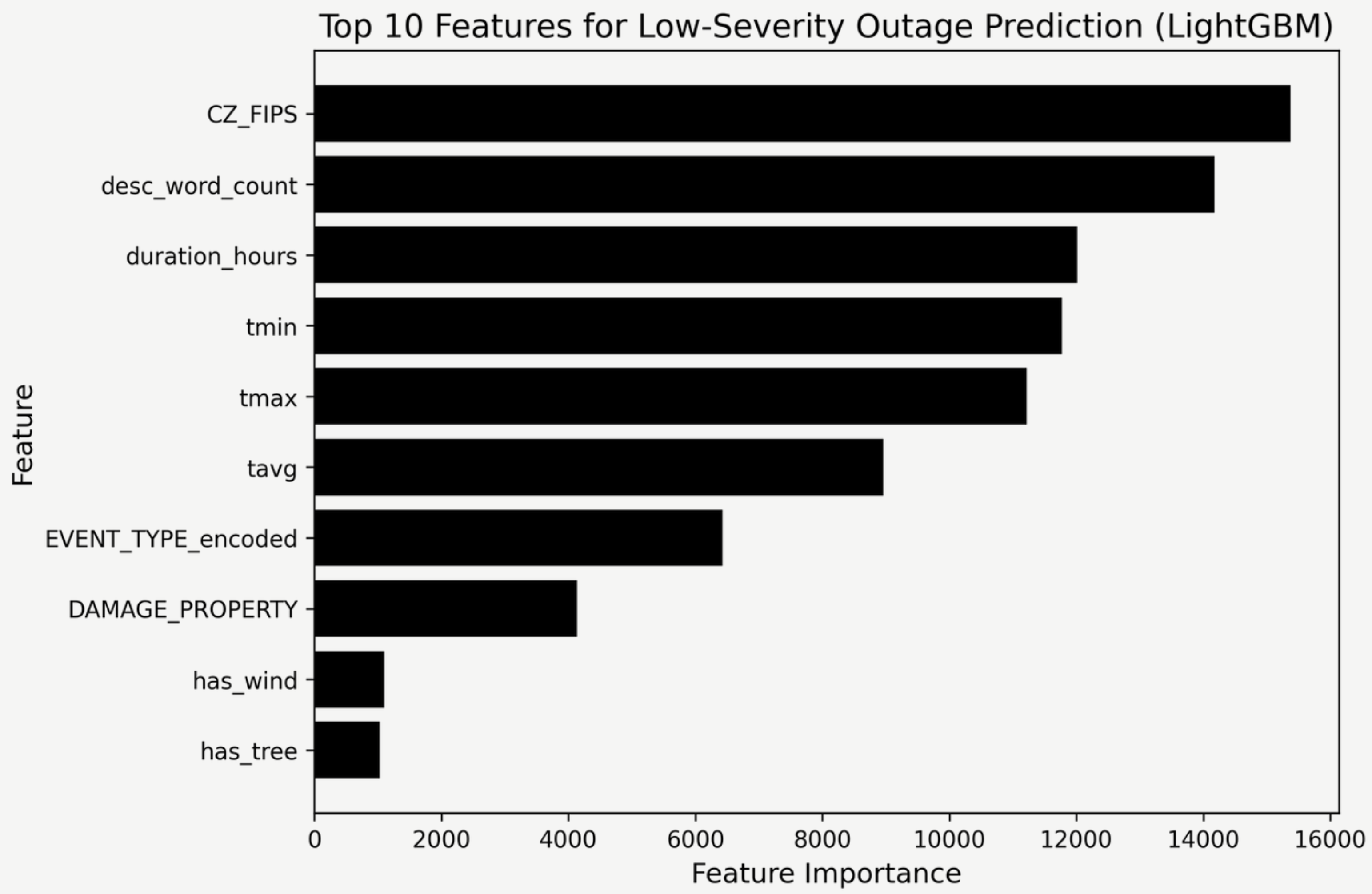
Outage Prediction - Low Severity

- **Task:**
  - Binary classification (is_outage: 0=no outage, 1=outage) for low-severity storms.
- **Model:**
  - LightGBM(n_estimators = 319, learning_rate = 0.047847333909262976, num_leaves = 281, max_depth = 20, min_child_samples = 22, subsample = 0.527136639688917, colsample_bytree = 0.8326768083509417)
- **Why LightGBM?:**
  - High performance with 19 features, including encoded variables.
  - Optimized for categorical and numerical feature interactions.
- **How It Works:**
  - Uses histogram-based learning for efficiency.
  - Builds trees leaf-wise, focusing on high-error regions.



Classification Report:

	precision	recall	f1-score	support
0	1.00	0.98	0.99	6916
1	0.98	1.00	0.99	6603
accuracy			0.99	13519
macro avg	0.99	0.99	0.99	13519
weighted avg	0.99	0.99	0.99	13519



Medium Severity

Random Forest

```
rf_model = RandomForestClassifier(
    n_estimators=229,
    max_depth=38,
    min_samples_split=5,
    min_samples_leaf=1,
    max_features='log2',
    bootstrap=False,
    random_state=42,
    n_jobs=-1
)
rf_model.fit(X_train_scaled, y_train)
```

→

```
Accuracy: 0.9583
Precision (weighted): 0.9590
Recall (weighted): 0.9583
F1 Score (weighted): 0.9583
```

XGBoost

```
best_params = {
    'n_estimators': 284,
    'max_depth': 15,
    'learning_rate': 0.07917066358756152,
    'subsample': 0.8015700617998658,
    'colsample_bytree': 0.5290123884596338,
    'gamma': 0.03854854253333355,
    'reg_alpha': 0.13733556293746432,
    'reg_lambda': 0.9769107444083599,
    'random_state': 42,
    'use_label_encoder': False,
    'eval_metric': 'logloss',
    'n_jobs': -1
}
```

→

```
Accuracy: 0.9601
Precision (weighted): 0.9609
Recall (weighted): 0.9601
F1 Score (weighted): 0.9601
```

LightGBM

```
lgb_model = lgb.LGBMClassifier(
    n_estimators=542,
    learning_rate=0.10968121619531485,
    num_leaves=119,
    max_depth=18,
    min_child_samples=16,
    subsample=0.8425576982327013,
    colsample_bytree=0.7109561164593746,
    reg_alpha=0.6242839973893298,
    reg_lambda=0.00655571732596838,
    random_state=42,
    n_jobs=-1
)
```

→

```
Accuracy: 0.9581
Precision (weighted): 0.9588
Recall (weighted): 0.9581
F1 Score (weighted): 0.9581
```

CatBoost

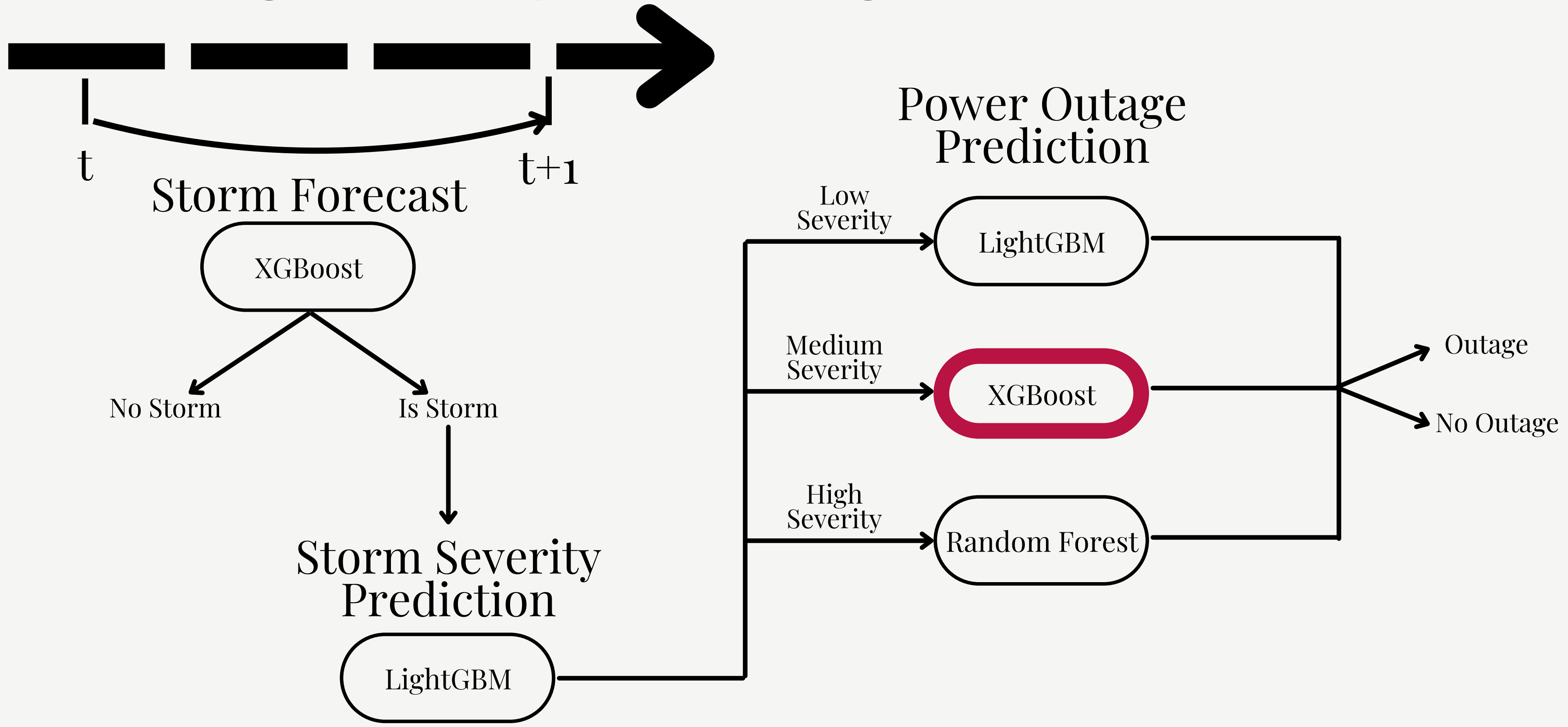
```
best_params = {
    'iterations': 562,
    'depth': 10,
    'learning_rate': 0.2194699455508984,
    'l2_leaf_reg': 1.6983857198493186,
    'border_count': 238,
    'random_strength': 0.5475461128526853,
    'bagging_temperature': 0.37619138326115115,
    'verbose': 0,
    'random_seed': 42
}
final_model = CatBoostClassifier(**best_params)
final_model.fit(X_train_scaled, y_train)
```

→

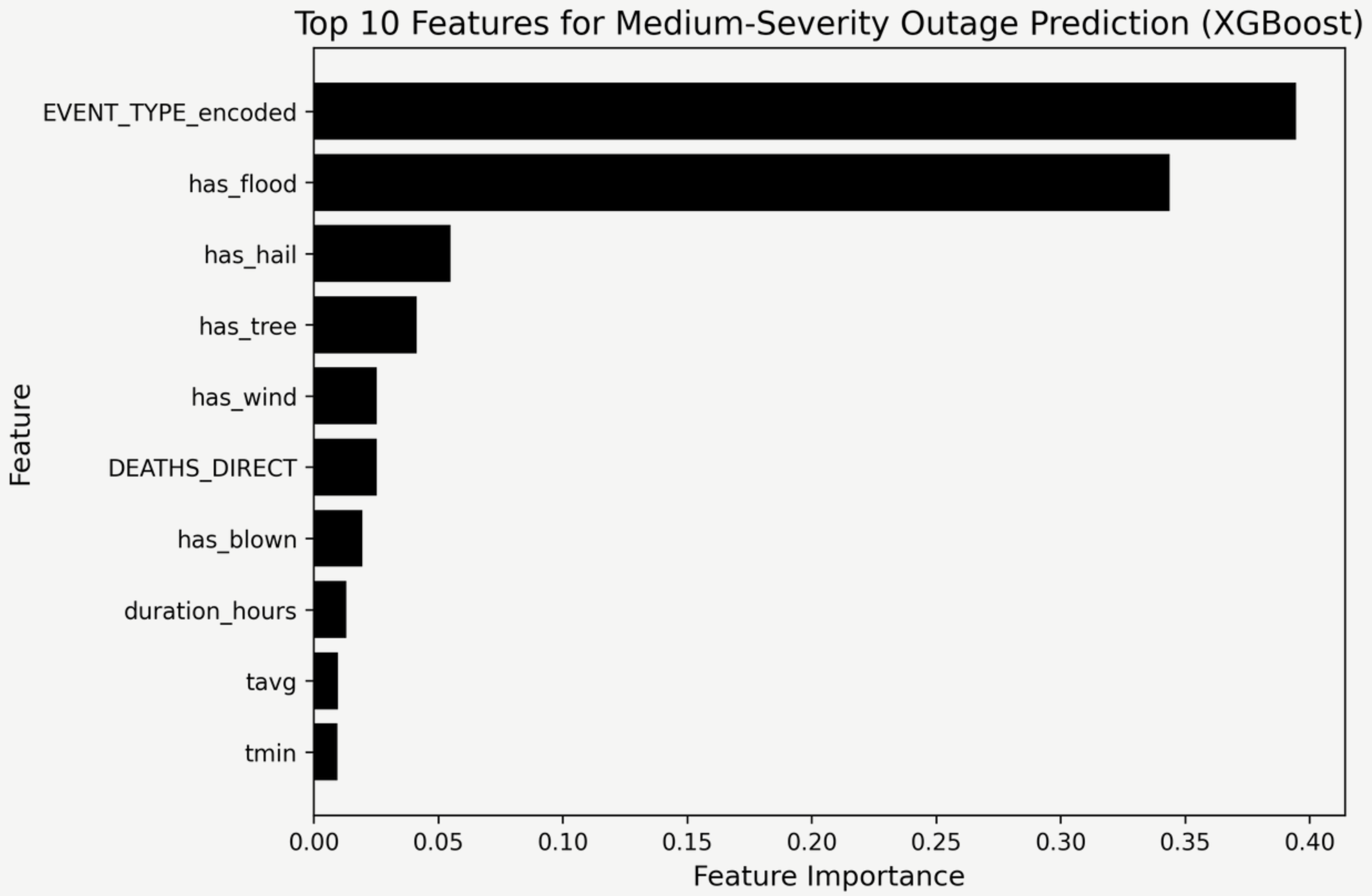
```
Accuracy: 0.9584
Precision (weighted): 0.9591
Recall (weighted): 0.9584
F1 Score (weighted): 0.9585
```

Outage Prediction - Medium Severity

- **Task:**
  - Binary classification (is_outage) for medium-severity storms.
- **Model:**
  - XGBoost(n_estimators=284, max_depth=15, learning_rate=0.07917066358756152, subsample=0.8015700617998658, colsample_bytree=0.5290123884596338, gamma=0.03854854253333355, reg_alpha=0.13733556293746432, reg_lambda=0.9769107444083599, random_state=42, use_label_encoder=False, eval_metric='logloss', n_jobs=-1)
- **Why Random Forest?:**
  - Excels in capturing feature interactions (19 features).
  - Handles class imbalance effectively with weighted loss.
- **How It Works:**
  - Iteratively builds trees, minimizing a loss function.
  - Uses regularization to prevent overfitting.




```
Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.96      0.98      6898
           1       0.96      0.99      0.97      6285

    accuracy                           0.97     13183
   macro avg       0.97      0.98      0.97     13183
weighted avg       0.98      0.97      0.97     13183
```



High Severity

Random Forest

```
rf_model = RandomForestClassifier(
    n_estimators=201,
    max_depth=38,
    min_samples_split=4,
    min_samples_leaf=1,
    max_features='sqrt',
    bootstrap=False,
    random_state=42,
    n_jobs=-1
)
```

→

```
Accuracy: 0.9656
Precision (weighted): 0.9664
Recall (weighted): 0.9656
F1 Score (weighted): 0.9657
```

XGBoost

```
best_params = {
    'n_estimators': 259,
    'max_depth': 15,
    'learning_rate': 0.09299367597135838,
    'subsample': 0.8143647474866939,
    'colsample_bytree': 0.7911741209414708,
    'gamma': 0.41166113452377523,
    'reg_alpha': 0.7767757103621074,
    'reg_lambda': 0.20469230843887554,
    'random_state': 42,
    'use_label_encoder': False,
    'eval_metric': 'logloss',
    'n_jobs': -1
}
```

→

```
Accuracy: 0.9626
Precision (weighted): 0.9637
Recall (weighted): 0.9626
F1 Score (weighted): 0.9627
```

LightGBM

```
best_params = {
    'n_estimators': 248,
    'max_depth': 12,
    'learning_rate': 0.21777108037853535,
    'num_leaves': 189,
    'feature_fraction': 0.6025416905004299,
    'bagging_fraction': 0.9011328465038242,
    'bagging_freq': 3,
    'lambda_l1': 1.3085313154773883,
    'lambda_l2': 2.017680534793574,
    'objective': 'binary',
    'metric': 'binary_logloss',
    'verbosity': -1,
    'boosting_type': 'gbdt',
    'random_state': 42,
    'n_jobs': -1
}
final_model = lgb.LGBMClassifier(**best_params)
final_model.fit(X_train_scaled, y_train)
```

→

```
Accuracy: 0.9619
Precision (weighted): 0.9626
Recall (weighted): 0.9619
F1 Score (weighted): 0.9619
```

FNN

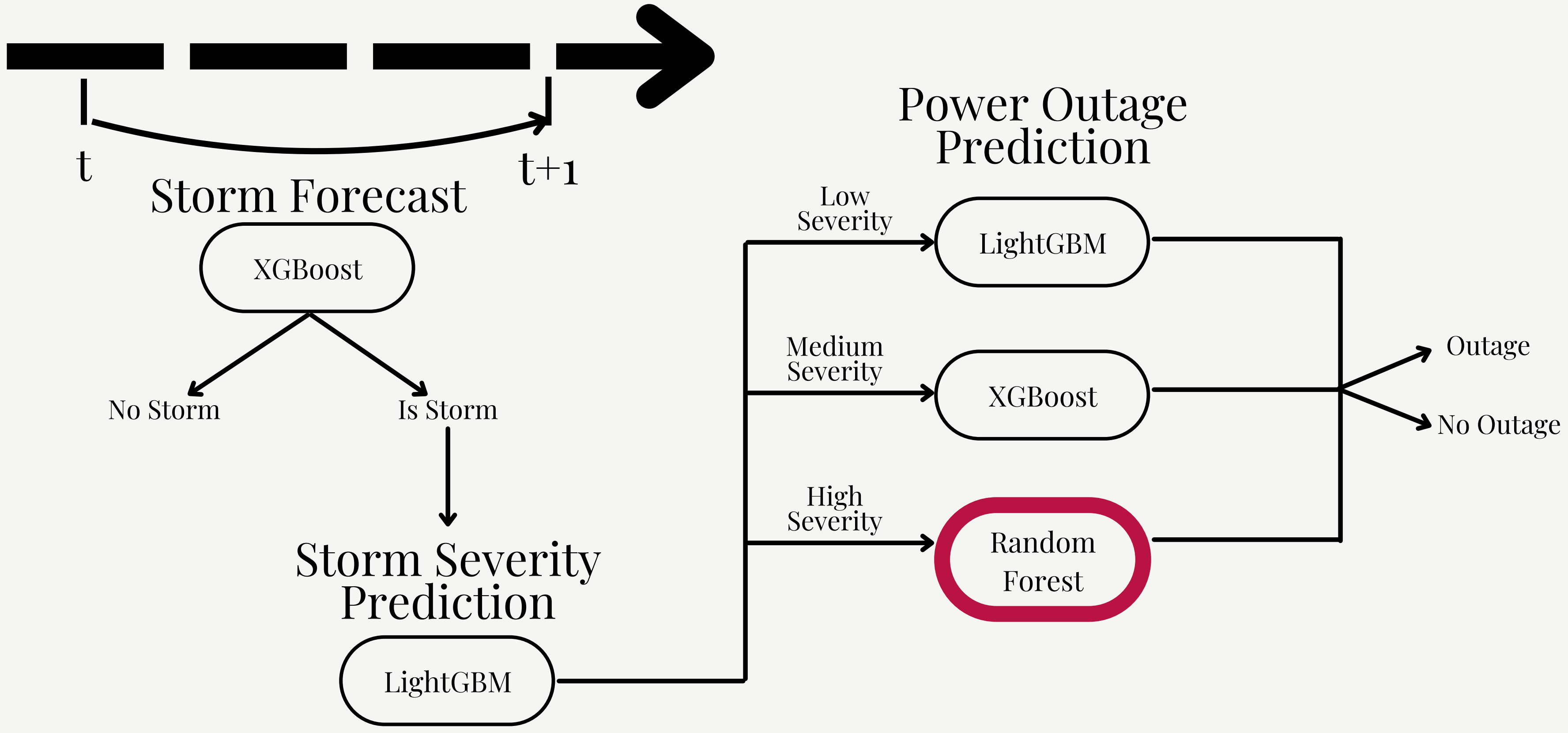
```
n_units = [80, 112, 96]
dropout_rate = 0.1677425146431672
learning_rate = 0.0017153835874558835
batch_size = 64

model = Sequential()
model.add(Dense(n_units[0], activation='relu
model.add(Dropout(dropout_rate))
model.add(Dense(n_units[1], activation='relu
model.add(Dropout(dropout_rate))
model.add(Dense(n_units[2], activation='relu
model.add(Dense(1, activation='sigmoid'))
```

→

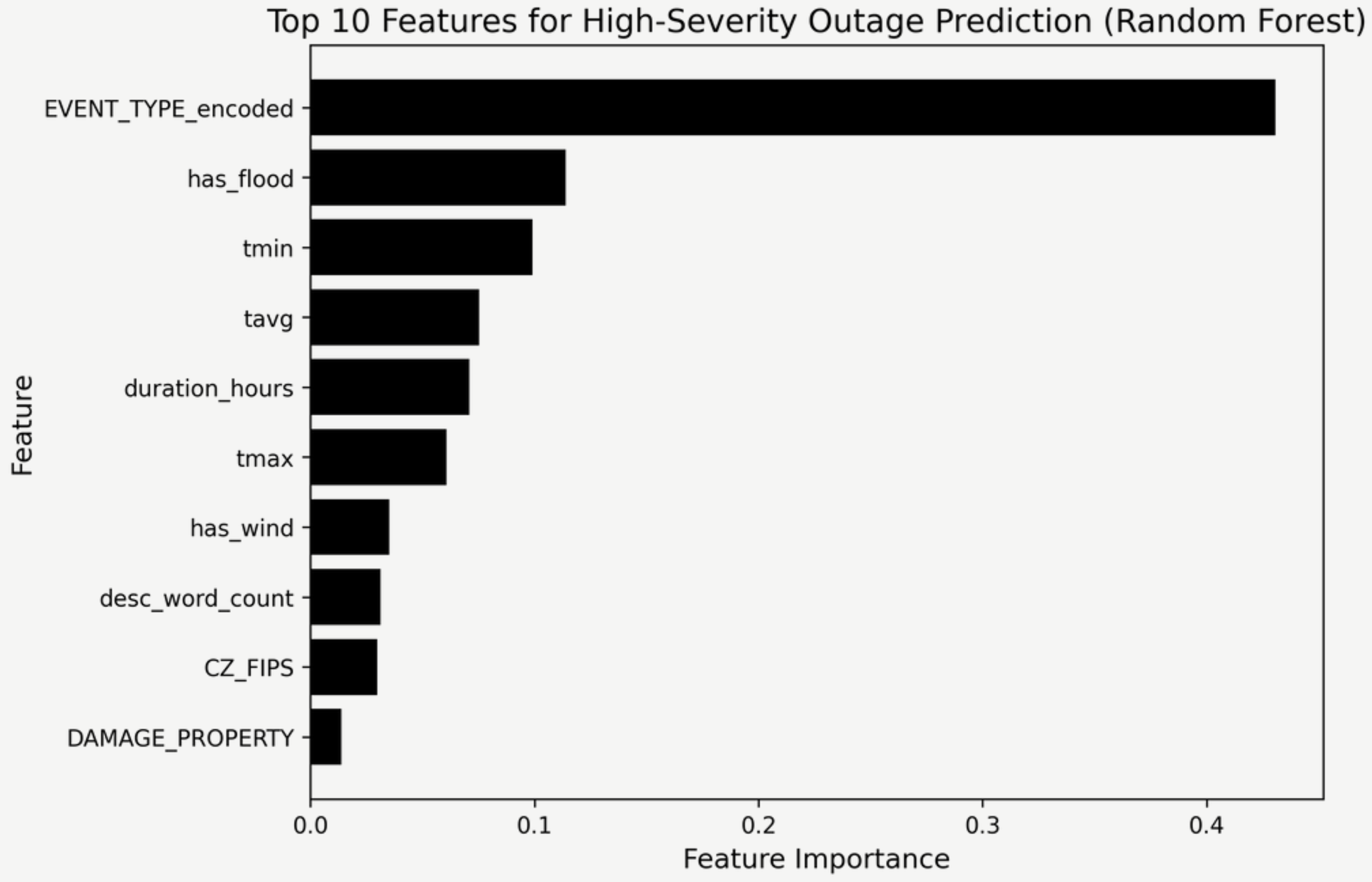
```
Accuracy: 0.9382
Precision (weighted): 0.9438
Recall (weighted): 0.9382
F1 Score (weighted): 0.9382
```

Outage Prediction - High Severity

- **Task:**
  - Binary classification (is_outage) for high-severity storms.
- **Model:**
  - RandomForestClassifier (n_estimators=201, max_depth=38, min_samples_split=4, min_samples_leaf=1, max_features='sqrt', bootstrap=False, random_state=42, n_jobs=-1).
- **Why Random Forest?:**
  - Robust to noise in high-severity data (19 features).
  - Reduces overfitting via ensemble of decision trees.
- **How It Works:**
  - Constructs multiple decision trees on bootstrapped data.
  - Aggregates predictions via majority voting.




```
Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.96      0.98      6910
           1       0.96      0.99      0.98      6662

    accuracy                           0.98     13572
   macro avg       0.98      0.98      0.98     13572
weighted avg       0.98      0.98      0.98     13572
```



Challenges and Solutions

***Challenge*: Data imbalance due to missing outage records not caused by storms**

Solution: Included non-storm-related outage data and ensured balanced distribution across all three severity levels to prevent overfitting

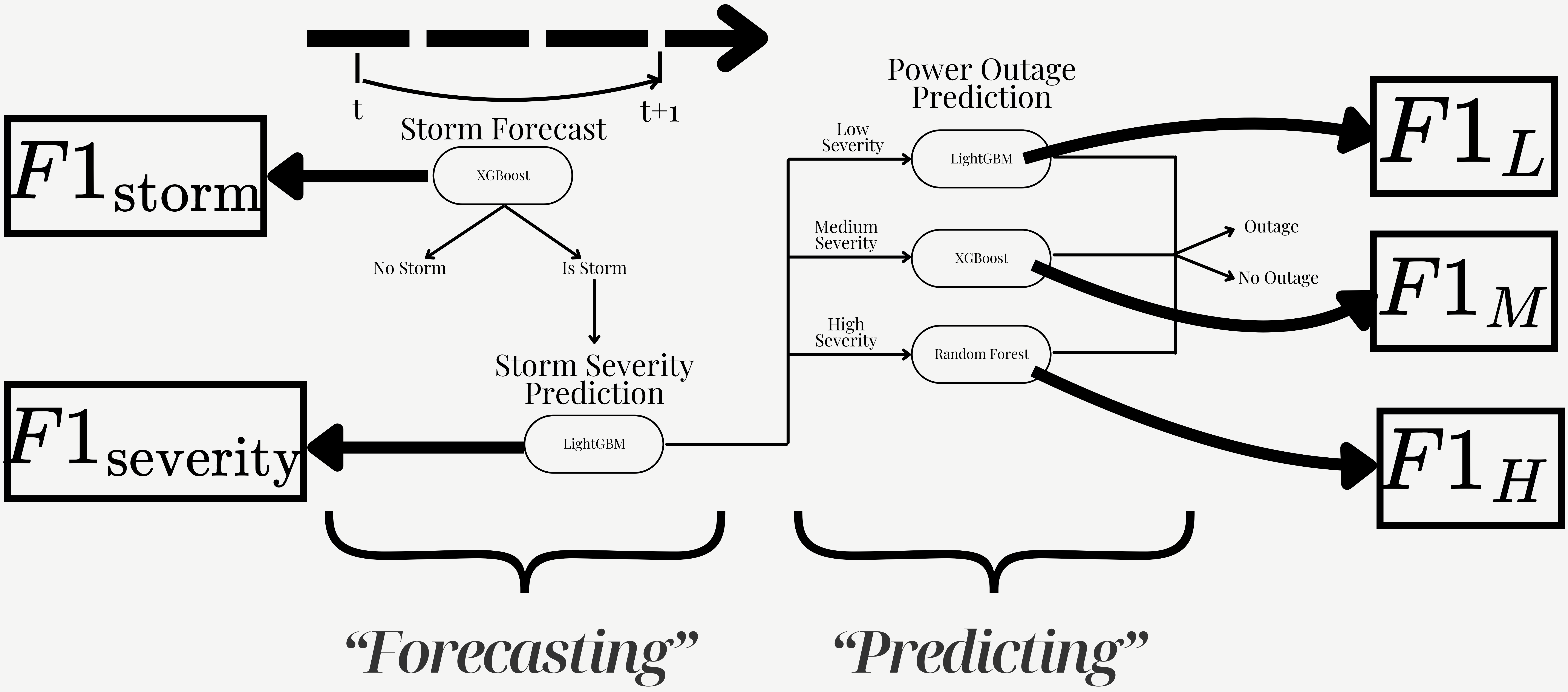
***Challenge*: Existing model evaluation metrics were incompatible with our methodology.**

Solution: Developed custom performance metrics by aggregating F1 scores across individual processing steps to accurately assess model effectiveness.

***Challenge*: Model Performance Stagnating with existing features**

Solution: Integrated Natural Language Processing (NLP) techniques to extract richer insights from text data, resulting in a ~1.5x improvement in model performance.

Performance Metric



Performance Metric

Performance for Forecasting

$$F1_{\text{storm}} \cdot F1_{\text{severity}} = 0.8683$$

Performance for Predicting

$$w_L \cdot F1_L + w_M \cdot F1_M + w_H \cdot F1_H = 0.98$$

$$\text{where} \quad w_L = \frac{n_L}{n_L + n_M + n_H}, \quad w_M = \frac{n_M}{n_L + n_M + n_H}, \quad w_H = \frac{n_H}{n_L + n_M + n_H}$$

Overall Performance Metric

$$\boxed{M_{\text{final}} = F1_{\text{storm}} \cdot F1_{\text{severity}} \cdot (w_L \cdot F1_L + w_M \cdot F1_M + w_H \cdot F1_H)}$$

$$= 0.8503$$

Why this Performance Metric?

$$\boxed{M_{\text{final}} = F1_{\text{storm}} \cdot F1_{\text{severity}} \cdot (w_L \cdot F1_L + w_M \cdot F1_M + w_H \cdot F1_H)}$$

- **Why F1 Score?**
  - F1-scores address class imbalance (e.g., rare "Is Storm" or "Outage" cases), making sure that the metric isn't skewed by majority classes, unlike accuracy.
  - F1-score (harmonic mean of precision and recall) penalizes false positives and false negatives equally. For example, in Outage Prediction, a false positive (predicting an outage when none occurs) might trigger unnecessary alerts, while a false negative (missing an outage) could leave systems unprepared — both are equally costly, so F1 balances these errors.
- **Taking into account error propagation:**
  - Multiplying F1_storm and F1_severity reflects the pipeline's sequential nature, where errors in storm detection or severity classification propagate and impact outage predictions.
- **Weighted Outage Predictions**
  - $w_L \cdot F1_L + w_M \cdot F1_M + w_H \cdot F1_H$ balances the contributions of Low, Medium, and High severity branches based on their frequency, ensuring fair representation of each severity level's performance.

Benchmarking Our Model

Forecasting Benchmark

Model	Task	Metric	Value
Our	Storm Presence Classification	F1-score	**0.9308**
Our	Storm Severity Classification	F1-score	**0.9329**
Our	Combined Forecasting Metric	F1{storm} × F1{severity}	**0.8683**
RAIN-F+ (2024)	Precipitation Classification (>0.1 mm/h)	F1-score	~0.70
Saito et al. (2018)	Tropical Cyclone Detection	POD (Probability of Detection)	0.912
Yang et al. (2020)	Storm Severity Classification	Accuracy	0.76
Lagerquist et al. (2020)	Convective Storm Classification	Not specified	-

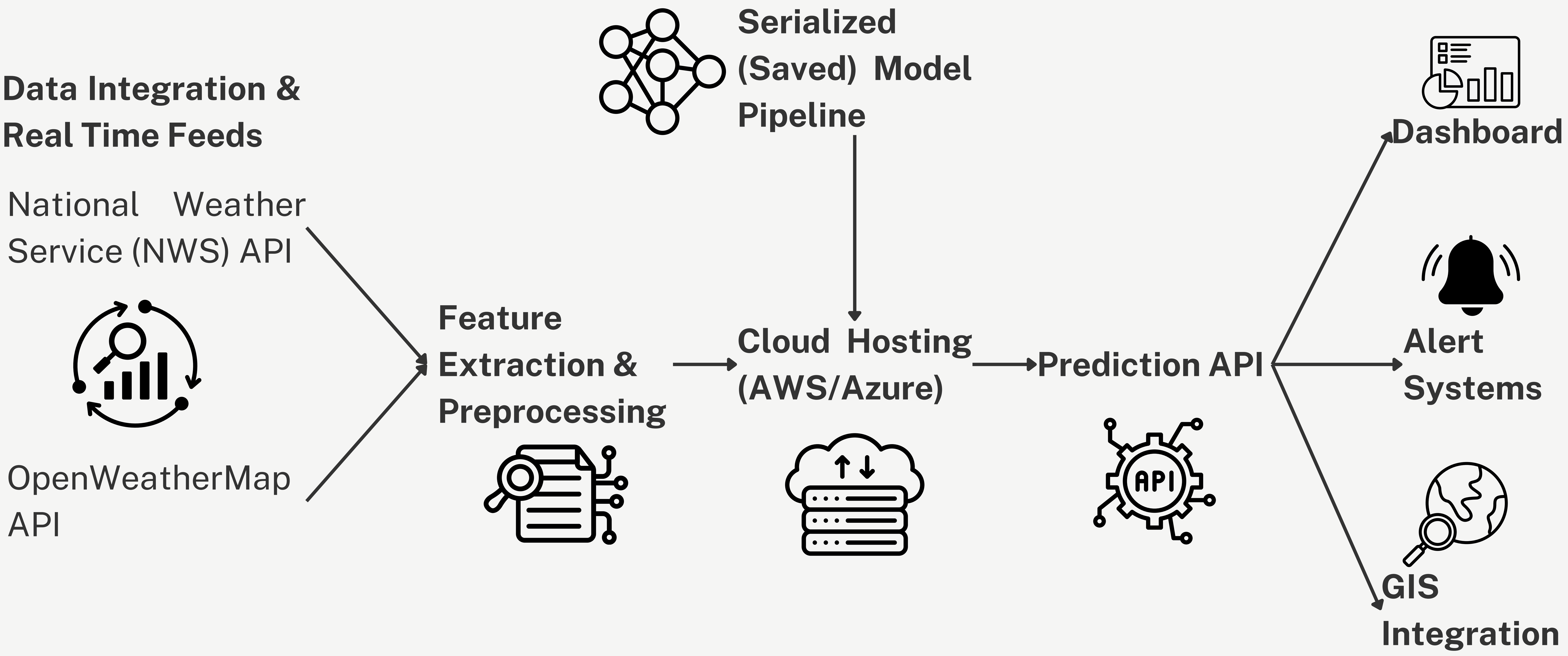
Predicting Benchmark

Model	Task	Metric	Value
Our	Power Outage Prediction (weighted across severity)	Weighted F1-score	**0.98**
Ertekin et al. (2024)	Power Outage Duration Prediction	Accuracy	0.98433
Ertekin et al. (2024)	Power Outage Duration Prediction	F1-score	0.98449
Khodaei (2016)	Component Outage Prediction	F1-score	0.9027
Wang et al. (2024)	Outage Probability Prediction	MAE	0.01346–0.03547
Mohammadi et al. (2020)	Power Line Outage Identification	Not specified	-

Sequential Forecasting and Prediction Models Comparison

Model	Tasks	Metrics	Values
Our	Storm Forecasting + Severity Classification + Outage Prediction	Combined Forecasting Metric	**0.8503**
Yang et al. (2020)	Severity Classification + Outage Prediction	Severity Classification Accuracy	0.76
Cerrai et al. (2019)	Weather Forecasting + Outage Prediction	Not specified	-

Deployability of Our Model in Future



Challenges Deployability Can Face

- **Data Latency and Availability**
  - Delays or missing real-time weather data during storms can lead to late predictions, reducing response time for utilities.
- **Model Drift and Generalization**
  - Shifting weather patterns or regional differences can degrade model accuracy, requiring frequent retraining.
- **Handling Noisy or Incomplete Data**
  - Noisy (e.g., erroneous wind speed) or incomplete (e.g., missing storm path) data can cause incorrect classifications, cascading errors through the pipeline.
- **Regulatory and Ethical Concerns**
  - Errors in predictions may lead to legal or ethical issues, such as neglecting regions due to false negatives, requiring compliance and fairness checks.

References

1. Cerrai, D., Wanik, D. W., Bhuiyan, M. A. E., Zhang, X., & Yang, F. (2019). Enhancing power outage prediction by integrating social media sentiment analysis. 2019 IEEE Power & Energy Society General Meeting (PESGM), 1–5. IEEE. https://www.researchgate.net/publication/331460438_Enhancing_Power_Outage_Prediction_by_Integrating_Social_Media_Sentiment_Analysis
2. Ertekin, Ş., Özbey, N., & Güngör, M. (2024). Machine learning model development to predict power outage duration (POD): A case study for electric utilities. Sensors, 24(13), 4313. https://doi.org/10.3390/s24134313
3. Essa, Y., Oudah, H. K., & Balanuta, A. (2022). Deep learning prediction of thunderstorm severity using remote sensing weather data. 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), 0645–0651. IEEE. https://www.researchgate.net/publication/360416940_Deep_Learning_Prediction_of_Thunderstorm_Severity_Using_Remote_Sensing_Weather_Data
4. Khodaei, A. (2016). Machine learning based power grid outage prediction in response to extreme events. 2016 North American Power Symposium (NAPS), 1–6. IEEE. https://www.researchgate.net/publication/310739737_Machine_Learning_Based_Power_Grid_Outage_Prediction_in_Response_to_Extreme_Events
5. Lagerquist, R., McGovern, A., & Smith, T. (2020). Classifying convective storms using machine learning. Weather and Forecasting, 35(2), 537–559. https://doi.org/10.1175/WAF-D-19-0096.1
6. Mohammadi, F., Zheng, C., & Su, R. (2020). Machine learning methods for power line outage identification. Electric Power Systems Research, 188, 106576. https://doi.org/10.1016/j.epsr.2020.106576
7. Saito, K., Takemi, T., & Mori, N. (2018). Deep learning approach for detecting tropical cyclones and their precursors in the simulation by a cloud-resolving global nonhydrostatic atmospheric model. Progress in Earth and Planetary Science, 5(1), 80. https://doi.org/10.1186/s40645-018-0245-y
8. Wang, Y., Cerrai, D., Anagnostou, E., & Wanik, D. (2024). Deep learning-based weather-related power outage prediction with socio-economic and power infrastructure data. arXiv. https://arxiv.org/abs/2404.03115 (Original work published April 4, 2024)
9. Yang, F., Watson, P., Koukoula, M., & Anagnostou, E. (2020). Enhancing weather-related power outage prediction by event severity classification. 2020 IEEE Power & Energy Society General Meeting (PESGM), 1–5. IEEE. https://www.researchgate.net/publication/340181673_Enhancing_Weather-Related_Power_Outage_Prediction_by_Event_Severity_Classification
10. RAIN-F+ (2024): Li, J. M., Aytac, J., Johnson-Freyd, P., Ahmed, A., & Holtzen, S. (2024). A nominal approach to probabilistic separation logic. doi:10.48550/ARXIV.2405.06826
11. Apenteng, B. A., Opoku, S. T., Ansong, D., Akowuah, E. A., & Afriyie-Gyawu, E. (2018). The effect of power outages on in-facility mortality in healthcare facilities: Evidence from Ghana. Global Public Health, 13(5), 545–555. https://doi.org/10.1080/17441692.2016.1217031
12. Greenwald, P. W., Rutherford, A. F., Green, R. A., & Giglio, J. (2004). Emergency department visits for home medical device failure during the 2003 North America blackout. Academic Emergency Medicine: Official Journal of the Society for Academic Emergency Medicine, 11(7), 786–789. doi:10.1197/j.aem.2003.12.032
13. Macmillan, M., Wilson, K., Baik, S., Carvallo, J. P., Dubey, A., & Holland, C. A. (2023). Shedding light on the economic costs of long-duration power outages: A review of resilience assessment methods and strategies. Energy Research & Social Science, 99(103055), 103055. doi:10.1016/j.erss.2023.103055
14. Adhikari, B., Mishra, S. R., Babu Marahatta, S., Kaehler, N., Paudel, K., Adhikari, J., & Raut, S. (2017). Earthquakes, fuel crisis, power outages, and health care in Nepal: Implications for the future. Disaster Medicine and Public Health Preparedness, 11(5), 625–632. doi:10.1017/dmp.2016.195

Thank You