1) Existing Solutions (Literature Snapshot)
Below is a summary of prominent approaches relevant to flood forecasting. The list begins with established methods for arid regions (A–C) and proceeds to recent US-based advancements (D–F) specifically selected for their ability to tackle the "Generalization Problem."

A. Entity-Aware LSTM (EA-LSTM) for Ungauged Basins

- **Reference:** Kratzert, F., et al. (2019). Towards learning universal, regional, and local hydrological behaviors via machine learning.
- **Description of Method:** This is an advanced version of the standard LSTM (Long Short-Term Memory) network. It explicitly separates dynamic inputs (rain, temperature) from static inputs (soil type, basin slope). The static features are processed by a special "gate" that controls the model, effectively giving each basin a unique "hydrological personality" based on its physical attributes.
- **Where it works well:** Ungauged Basins. It excels at Transfer Learning. The model can learn how specific rock types react to rain in a monitored area (e.g., the North) and apply that logic to an unmonitored stream in the South with similar geology, without needing historical data for calibration.
- **Fails / Risks:** Physics Violation. Since it is a "black box," the model does not strictly obey physical laws (like conservation of mass). In extreme, unseen weather events, it might predict physically impossible water volumes.

B. Physics-Informed Graph Neural Networks (GNNs)

- **Reference:** Moshe, Z., et al. (2020). HydroNets: Leveraging River Structure for Hydrologic Modeling.
- **Description of Method:** This approach represents the river network as a mathematical Graph, where nodes are sub-basins and edges are river segments. It uses "Message Passing" to simulate water flowing from upstream to downstream nodes, mimicking the actual physical flow of the river.
- **Where it works well:** Complex Topologies. It is ideal for complex wadi networks (common in the Negev), where the timing of a flood depends heavily on the routing and connection between different tributaries. Standard models often miss this spatial context.
- **Fails / Risks:** Data Dependencies. The model relies on a perfect map of the river network. If the topological data (DEM) has errors—which is common in flat desert terrains—the model will route the water incorrectly.

C. Flash Flood Classification using GNSS Water Vapor (The Israeli Approach)

- **Reference:** Ziskin, Z., & Reuveni, Y. (2021). The prediction of flash floods... using GNSS-derived precipitable water vapor.
- **Description of Method:** A unique Israeli approach that uses Support Vector Machines (SVM). Instead of relying only on standard weather radar, it uses Precipitable Water Vapor (PWV) derived from ground-based GPS/GNSS stations to detect moisture build-up in the atmosphere before a storm.
- **Where it works well:** Overcoming Radar Blockage. This is critical for the Negev and Arava regions, where the IMS weather radar suffers from "beam

blockage" (hidden by mountains) and distance effects. GNSS provides a reliable signal regardless of radar visibility.

- **Fails / Risks:** No Temporal State. As a classical ML classifier, it lacks "memory." It can predict the probability of a flood well, but it cannot accurately predict the amount of water (hydrograph) or the exact timing compared to sequence models like LSTM.

Recent Solutions for the Generalization Problem (2024–2025)

Rationale for Selection: The following three articles were specifically chosen because they address the primary bottleneck in modern hydrological research: the **Generalization Problem**, often referred to as **PUB** (Prediction in Ungauged Basins). While traditional models perform well on familiar data, they struggle to generalize to new areas without sensors. These recent US-based studies propose distinct architectural shifts—Physical, Hybrid, and Spatial—to solve this challenge.

D. Physics-Informed Machine Learning (PIML) for Compound Mapping

- **Reference:** University of Alabama & NOAA (2024). Physics-Informed Machine Learning for Compound Flood Mapping.
- **Description of Method:** Addresses the generalization problem by embedding differential equations (such as Saint-Venant flow equations) directly into the neural network's loss function. Instead of learning solely from statistics, the model is constrained by the laws of physics.
- **Where it works well:** Zero-Shot Generalization. Because the laws of physics are universal, this model allows for reliable predictions in areas with zero historical data, preventing the "hallucinations" common in standard deep learning models when facing new environments.
- **Fails / Risks:** Complexity. Training these models is computationally expensive and requires complex mathematical formulation compared to standard data-driven models.

E. Hybrid AI Error Correction Framework

- **Reference:** University of Michigan & NOAA (2025). AI boosts National Weather Model flood prediction accuracy sixfold.
- **Description of Method:** A hybrid architecture where AI does not replace the traditional physical model but acts as a post-processing "corrector." The AI learns the specific error patterns of the physical model (National Water Model) and corrects them in real-time.
- **Where it works well:** Extreme Events. This approach has shown significant improvement (up to 6x accuracy) in predicting extreme flood events, which are traditionally the hardest to generalize for. It combines the spatial stability of physical models with the adaptability of AI.
- **Fails / Risks:** Dependency. The system is entirely dependent on the availability and stability of the underlying physical model.

F. Spatiotemporal Transfer Learning

- **Reference:** PLOS ONE (2025). Hydrological prediction in ungauged basins based on spatiotemporal characteristics.
- **Description of Method:** A pure Transfer Learning approach that maps static geographical features (slope, soil, vegetation) to hydrological responses. It trains on data-rich basins (like the CAMELS dataset) to learn universal rules of how geography dictates water flow.
- **Where it works well:** Practical Implementation. It offers the most straightforward engineering solution for the "Prediction in Ungauged Basins" (PUB) problem, allowing logic learned in one part of the world to be applied to another with similar topography.

- **Fails / Risks:** Climatic Variance. It assumes that the relationship between geography and flow remains constant across different climatic zones, which may not always hold true.

Bonus: Critique of Ziskin & Reuveni (2021)
Identified Pitfall: Data Leakage via "Random Split"

While the study presents impressive accuracy (AUC 0.93), the validation methodology contains a significant flaw: the use of a randomized 80/20 train-test split on time-series data.

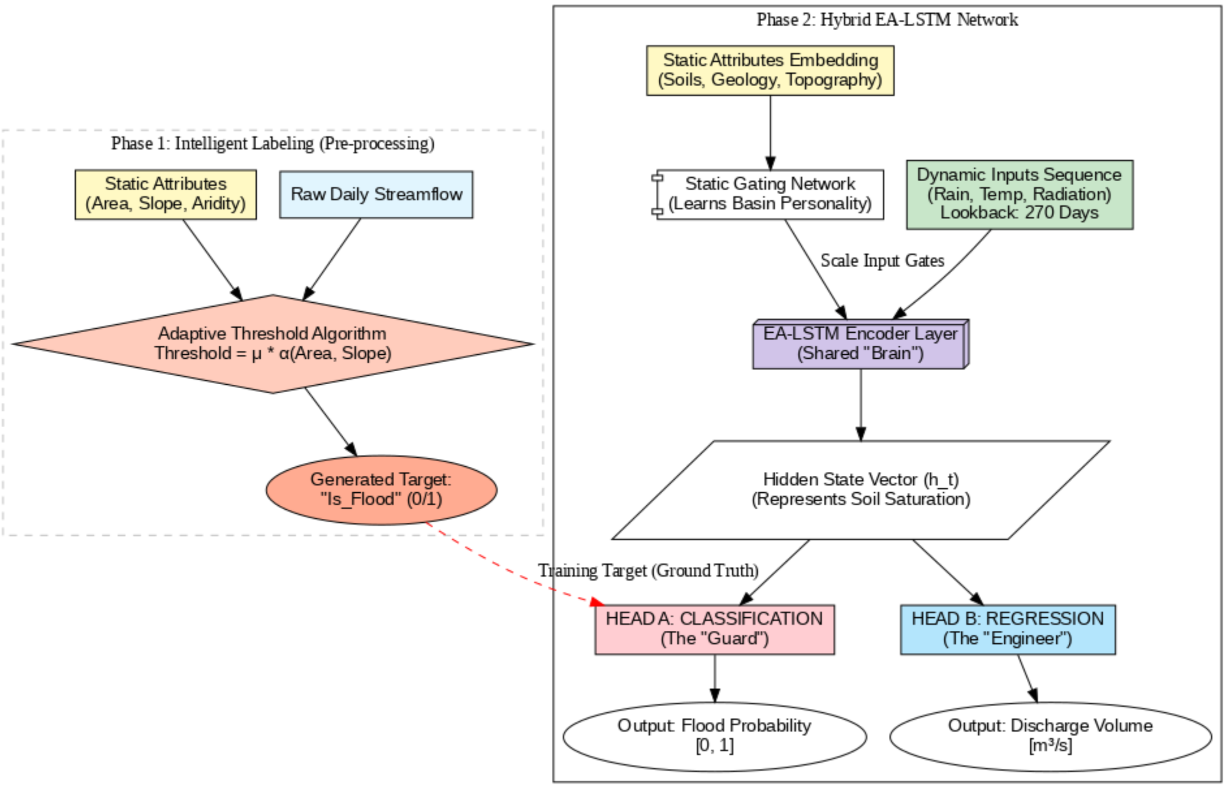
- **The Problem (Leakage):** Hydrological and meteorological data have high Autocorrelation (the weather at 12:00 is very similar to 13:00). By splitting the data randomly, the model likely sees data points from the same storm event in both the training set and the test set.
- **The Result:** The model is not learning to predict the future (Extrapolation); it is learning to interpolate missing hours within a known event. This leads to "Look-Ahead Bias" and over-optimistic results that will likely fail in a real-time operational setting where future data is not available.

Suggested Improvement: To prove the model works for real warnings, the evaluation must use a Temporal Split (e.g., Train on years 2015–2018, Test on 2019–2020) or a Leave-One-Event-Out approach. This ensures the model is tested on entirely new, unseen storms.

2. Proposed Methodology and Architecture

2.1 The Data Resolution Challenge & Labeling Strategy

A fundamental challenge in this study is the temporal resolution of the available data. The Dataset provides **Daily Mean Discharge**, whereas flash floods in arid catchments are often rapid, **sub-daily events**. Averaging a short, intense peak over 24 hours results in low numerical values that mask the event's severity. Consequently, a standard regression model trained on this smoothed data would likely converge to underestimate risks, interpreting low means as non-events.


Phase 1: Intelligent Labeling (Pre-processing)
Static Attributes (Area, Slope, Aridity)
Raw Daily Streamflow
Adaptive Threshold Algorithm Threshold = μ * α(Area, Slope)
Generated Target: "Is_Flood" (0/1)
Training Target (Ground Truth)
Phase 2: Hybrid EA-LSTM Network
Static Attributes Embedding (Soils, Geology, Topography)
Static Gating Network (Learns Basin Personality)
Dynamic Inputs Sequence (Rain, Temp, Radiation) Lookback: 270 Days
Scale Input Gates
EA-LSTM Encoder Layer (Shared "Brain")
Hidden State Vector (h_t) (Represents Soil Saturation)
HEAD A: CLASSIFICATION (The "Guard")
HEAD B: REGRESSION (The "Engineer")
Output: Flood Probability [0, 1]
Output: Discharge Volume [m³/s]

To overcome this, I developed a robust **Event Detection Strategy** rather than relying solely on raw discharge values.

- **Adaptive Thresholding:** I formulated a dynamic threshold index that accounts for static basin attributes. Since arid basins (e.g., Judean Desert) exhibit high flow variance (CV > 1), the threshold for defining a "flood" is adjusted relative to the basin's specific hydromorphology (Area, Slope) rather than a fixed global value.
- **Validation against Ground Truth:** Critically, this labeling logic was not theoretical. I performed a **validation process on a sample of Judean Desert streams** (e.g., Nahal Darga, Nahal Hemar), cross-referencing the daily data with external historical records (news archives, social media reports). This verified that even low daily means (e.g., 0.4 m^3/s) indeed corresponded to significant observed floods, justifying our sensitive labeling approach.

2.2 The Hybrid Multi-Task Architecture

We propose a **Hybrid Multi-Task EA-LSTM** (Entity-Aware LSTM) network. This architecture allows the model to learn simultaneously from two complementary objectives:

1. **The Shared Encoder (The "Brain"):** The core is an LSTM layer that processes meteorological forcing (Rain, Temp) over a **Lookback Window** of 270 days. This long window is crucial for capturing **Antecedent Soil Moisture** allowing the model to differentiate between rain on dry soil (infiltration) versus rain on saturated soil (runoff).

    - *Note:* To ensure applicability to Ungauged Basins (PUB), past discharge is strictly excluded from the inputs.
    - *Entity-Awareness:* Static attributes (Slope, Soil Type) are injected via a Gating Network, allowing the model to generalize physical laws across different catchments.

2. **The Dual-Head Output (The Synergy):** The encoder's output splits into two heads:
    - **Head A (Classification - Alert):** Predicts the probability of a flood event, P(Flood).
    - **Head B (Regression - Volume):** Predicts the daily discharge volume (Q).
    - *Synergy:* By optimizing both tasks together, the Classification head forces the shared encoder to recognize "flood patterns" even when the regression target is low due to averaging. This shared representation significantly improves the regression accuracy for extreme events.

2.3 Training Strategy to Prevent Overfitting

Given the relatively small dataset and the rarity of floods (imbalanced data), we employ strict regularization:

- **Composite Loss Function:**

$$\boldsymbol{L\_Total = L\_NSE\ (Regression) + \lambda * L\_Weighted - BCE\ (Classification).}$$

- We use **Weighted Binary Cross Entropy** for the classification head, assigning a high penalty weight to missed floods (False Negatives). This counteracts the data imbalance (95% dry days).
- **Spatial & Temporal Splitting:**
    - *Temporal:* Training and Testing sets are separated by years (e.g., Train: 1980-2010, Test: 2011-2020) to prevent future-peeking.
    - *Spatial Integrity:* Hydrologically connected basins (upstream/downstream) are kept in the same partition to prevent spatial data leakage.

- **Basin Dropout:** During training, static attributes are randomly masked. This forces the model to learn general physical relationships (e.g., "steep slope = fast response") rather than memorizing specific Basin IDs, ensuring robust generalization to unseen catchments.

3. Data Integrity and Integration Strategy

Mixing data from satellites (ERA5) and ground stations creates quality issues. We use specific methods to fix these problems and ensure the model learns real physics rather than technical errors.

Different Resolutions (Size Mismatch) Satellite pixels are very large, while some river basins are small. To fix this size mismatch, we average the satellite pixels over the basin area. Additionally, the model uses the **Static Embedding (Area)** to learn a "scaling factor," which helps it understand that rain on a huge pixel affects a small basin differently.

Time Alignment Satellites use global time (UTC), while local data might use local time, and a mismatch here is critical. We solve this by converting all data to **UTC+0**. We also verify that the "Daily Mean" covers the exact same hours in both sources, ensuring that rain always appears strictly before the flood.

Sensor Bias Satellites often underestimate rain in deserts, so the model might learn to "distrust" them. To fix this Sensor Bias, we use **Z-Score Normalization**. We don't feed raw numbers to the model; instead, we feed "how far is today's rain from the average," which cancels out the constant error of the satellite.

More critically, the dataset reveals that extreme rainfall events frequently coincide with zero or missing streamflow readings, **likely indicating sensor failures during actual floods.** These specific samples are "poisonous To handle this, we employ a **loss-masking strategy**: rather than removing data and breaking the temporal sequence, we apply a validity mask to these specific time steps during training. This ensures the loss function "ignores" these poisonous samples

Missing Data (Dual Strategy) We apply a strict distinction between input features and target labels. For missing **input features** (like temperature or radiation), we avoid simple, inaccurate methods like mean or median imputation. Instead, we use model-based imputation, utilizing the correlations between other available features to accurately predict and fill these gaps. However, for the **target variable** (discharge), we enforce a strict "No-Guessing" policy. We use a Mask to tell the loss function to ignore days with missing discharge records. We never impute the target because if we used a model to fill these gaps, our main model would simply learn to mimic those synthetic guesses rather than learning physical reality.

Noisy Data Measuring river flow is difficult and the numbers in the dataset can be noisy. Our **Multi-Task** approach helps solve this. Even if the exact volume number is inaccurate, the **Classification Head** ("Flood/No Flood") is stable and helps the model learn the correct patterns despite the noise.

Climate Change The climate changes over decades, so old data behaves differently than new data. We handle this by splitting the data by time. We train on the past and test strictly on **future years** (2011–2020). Also, the long memory (270 days) helps the model understand if the current season is dry or wet. However, **our inspection of the**

provided dataset reveals a significant challenge: prior to 1980, nearly all basins lack streamflow records (labels) despite having meteorological inputs. This specific data limitation forces us to discard decades of potential training signals and begin effective supervised training only from 1980 onwards.

4. Generalizing to Unseen Regions (Domain Shift Strategy)

Development Split: Grouped Spatial Cross-Validation To test generalization, we employ a Grouped Spatial Cross-Validation strategy. The fundamental constraint here is that hydrologically connected basins, such as an upstream tributary and its downstream outlet, are strictly kept within the same partition and are never separated between training and testing sets. This ensures that the Test set consists of strictly unseen basins that the model has never encountered during training. This approach simulates a true "Ungauged Basin" scenario, proving the model can generalize to new locations purely based on physics without having memorized their specific flow history.

Required Metadata: The "Normalization Key" To enable the model to function on these new basins, obtaining their static metadata—specifically Area, Slope, and Aridity Index **Crucially, these raw attributes are not fed directly; they are first projected via a dense layer into a learned static embedding vector.** Inside the network, this embedding acts as a "Normalization Key" that feeds the Gating Mechanism Inside the network, these static features feed the Gating Mechanism of the EA-LSTM, effectively normalizing the dynamic rainfall inputs by teaching the model that specific rainfall intensities behave differently depending on the basin's size and steepness. Furthermore, these metadata values are essential for the Adaptive Decision Threshold logic defined in our methodology. Since the cutoff for defining a flood is a function of the basin's attributes, we cannot calculate the specific prediction threshold for the new region without these static inputs.

5. Evaluation Plan and Validation Protocols

To ensure the results reflect true deployment performance and mitigate common validation traps in hydrological modeling, we employ the following rigorous evaluation framework:

A. Evaluation Metrics (Handling Imbalance & Regression) Since the model performs two distinct tasks, we evaluate them with separate metrics tailored to their specific objectives and data characteristics:

- **For Classification (The "Guard"): Prioritizing Safety** Standard Accuracy is a misleading metric due to the extreme class imbalance (>95% dry days). A trivial model predicting "No Flood" would achieve high accuracy but fail operationally. Therefore, we focus on:
    - **Recall (Sensitivity):** The primary metric for an early warning system. It measures the proportion of actual flood events correctly detected. Maximizing Recall is crucial to minimize life-threatening False Negatives.

    - **Precision & F1-Score:** Used to monitor false alarms. While high Recall is the priority, reasonable Precision is required to maintain user trust and avoid "alarm fatigue."

    - **AUC-PR (Area Under the Precision-Recall Curve):** The most informative aggregate metric for highly imbalanced datasets. Unlike standard accuracy or ROC-AUC (which can be misleadingly high when valid "non-flood" days dominate), AUC-PR focuses strictly on the minority class performance. It evaluates the model's global robustness across *all* possible decision thresholds, effectively summarizing the trade-off between catching floods and avoiding false alarms without committing to a single cutoff point.

- **For Regression (The "Engineer"): Hydrological Accuracy**
    - **NSE (Nash-Sutcliffe Efficiency):** The standard hydrological metric used to assess the predictive power relative to the observed variance. Unlike raw error metrics, NSE is normalized, allowing fair comparison across basins with vastly different discharge magnitudes.
    - **MSE/RMSE:** Used as auxiliary metrics to quantify the absolute volume error.

B. Temporal Validation Strategy (The "Time Trap") We explicitly reject random train test strategy strategies commonly used in other ML domains.

- **Why Random is Wrong:** Random splitting destroys the temporal sequence required by the LSTM (Lookback Window) and introduces severe data leakage due to the high autocorrelation of meteorological data (i.e., tomorrow's weather is highly correlated with today's).
- **Our Approach:** We employ a strict **Chronological Split**. The model is trained on historical data (e.g., 1980–2010) and evaluated on a strictly future block (e.g., 2011–2020). This simulates a real-world forecasting scenario where future data is unknown.

C. Spatial Generalization (The "Basin Holdout") To assess the model's ability to solve the PUB (Prediction in Ungauged Basins) problem, we perform **Spatial Validation**:

- **Unseen Basins:** We evaluate performance on a holdout set of basins that were never seen during training. This tests whether the model has learned generalizable physical laws (via the Static Embeddings) rather than memorizing the hydrographs of specific training catchments.
- **Regional Diversity:** Validation includes basins from diverse climatic zones (e.g., testing on arid southern basins after training on semi-arid northern ones) to verify robustness against domain shifts.

D. Hyperparameter Tuning To select the optimal configuration (e.g., Learning Rate, Hidden Size, Dropout), we perform **K-Fold Cross-Validation** strictly within the Training period. The Test set remains completely locked ("in the vault") during this process to prevent overfitting to the evaluation data.

6) Bonus: Minimal Implementation The complete proof-of-concept implementation, including the sequence model training pipeline, custom splitting strategy, and performance analysis, is provided in the accompanying Jupyter Notebook. Please refer to the attached notebook for the code, evaluation plots, and detailed inline explanations regarding generalization risks.