Region-Adaptive AI Framework for Landslide Risk Prediction Using Multi-Temporal Rainfall Data

Anupam Sharma
Assistant Professor
Department of Computer Science & Engineering
Chandigarh University, Mohali, India
anupam.e17312@cumail.in

Arun
Student
Department of Computer Science & Engineering
Chandigarh University, Mohali, India
kumararunlamba89@gmail.com

Yash Raj
Student
Department of Computer Science & Engineering
Chandigarh University, Mohali, India
yashraj.ap6045@gmail.com

Armaan Siag
Student
Department of Computer Science & Engineering
Chandigarh University, Mohali, India
armaansiag06@gmail.com

***Abstract*—Landslides are one of those disasters that don't always make headlines, but when they do, the damage is often devastating—especially in mountainous regions like the Himalayas or the Western Ghats. In most cases, heavy or prolonged rainfall quietly builds up pressure within the soil until, suddenly, the slope gives way. Predicting such events is not straightforward, mainly because rainfall behaves differently across regions, and terrain conditions vary just as much.**

In this work, we propose a Region-Adaptive Artificial Intelligence Framework (RAAF) that attempts to handle this complexity more realistically. Instead of relying on a single snapshot of rainfall, the model looks at how rainfall evolves over time—capturing short-term bursts as well as longer-term accumulation. To do this, we combine Convolutional Neural Networks (CNN) for understanding spatial terrain patterns with Long Short-Term Memory (LSTM) networks that learn how rainfall changes over days or even weeks.

One practical challenge we focused on is that models trained in one region often fail in another. To address this, we incorporate a transfer learning mechanism that allows the model to adapt using limited data from new regions.

Our results show that combining multi-temporal rainfall data with region-adaptive learning improves prediction accuracy to over 91

***Index Terms*—Landslide Prediction, Deep Learning, CNN-LSTM, Multi-Temporal Rainfall, Transfer Learning, Disaster Management**

I. Introduction

Landslides don't usually arrive with a warning siren. In many cases, they build up quietly—after days of steady rainfall, maybe a week of damp soil, sometimes even longer. Then, at a tipping point, the slope fails[6]. This is what makes them particularly difficult to predict and, frankly, quite dangerous in regions like the Himalayas, the Western Ghats, and several parts of Southeast Asia[2].

Over the years, researchers have tried to understand this behavior using different approaches. Traditional models—especially physically-based ones—attempt to simulate how water moves through soil and how slopes respond[5]. These models are detailed and scientifically sound, but they demand a level of data (like subsurface soil properties) that is rarely available, especially at large regional scales[8]. On the other hand, simpler statistical models rely on rainfall thresholds, such as how much rain falls within a certain duration. While these are easier to apply, they often oversimplify reality[7] and struggle to adapt across different terrains.

This is where machine learning entered the picture. Models like Random Forest and Support Vector Machines improved prediction accuracy by learning patterns directly from data rather than relying on fixed assumptions. Still, there's a catch—most of these models treat inputs as static snapshots. They don't fully capture how rainfall accumulates over time, which, as field studies repeatedly show, plays a crucial role in triggering landslides.

Deep learning models, particularly Long Short-Term Memory (LSTM) networks, changed that to some extent. They can "remember" past rainfall and learn temporal dependencies. When combined with Convolutional Neural Networks (CNN), which are good at extracting spatial features like terrain patterns, the result is a more holistic understanding of the problem. But even with this progress, one major issue remains unresolved[10].

A model trained in one region often struggles when applied somewhere else.

And it makes sense when you think about it. The geology of Yunnan in China is not the same as the Western Ghats in India. Rainfall patterns, vegetation cover, soil composition—everything shifts[13]. So expecting one model to work everywhere without adaptation is. . . a bit unrealistic.

This paper builds on that realization.

We propose a Region-Adaptive Artificial Intelligence Framework (RAAF) that not only models rainfall over multiple time scales but also adapts itself to different geographic regions. The idea is simple in principle: learn general patterns from data-rich regions and then fine-tune the model for regions where data is limited.

The overall workflow of the proposed system is illustrated in Fig. 1. It shows how raw environmental data moves through preprocessing, feature extraction, deep learning prediction, and finally into risk classification[12].

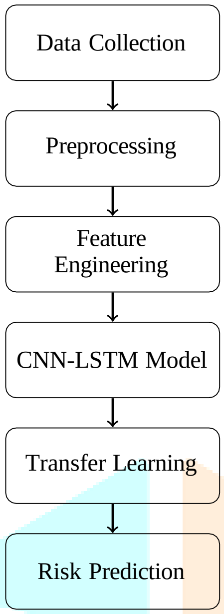

Fig. 1: Overall workflow of the proposed system

At the core of the framework is the use of multi-temporal rainfall features. Instead of relying on a single rainfall value, we consider rainfall accumulated over different durations—short-term (3-day), medium-term (7-day), and longer-term (30-day). This allows the model to capture both immediate triggers and gradual soil saturation effects[1].

To give a clearer idea of the inputs used in the system, Table I summarizes the key features considered in this study.

TABLE I: Key Input Features

Feature	Description
Rainfall (24h)	Instantaneous rainfall (mm)
Rainfall (7-day)	Weekly accumulated rainfall
Rainfall (30-day)	Monthly accumulated rainfall
Slope Gradient	Derived from DEM
Soil Moisture	Saturation level (%)
NDVI	Vegetation index

Putting it all together, the goal of this research is not just to improve prediction accuracy, but to make the model more adaptable and practical for real-world use—especially in regions where data is limited but the risk is high.

In the following sections, we first review existing work in landslide prediction, then describe the proposed framework in detail, followed by experimental evaluation and discussion.

II. Literature Review

When you start digging into landslide prediction research, one thing becomes clear pretty quickly—there isn't a single "best" method. Instead, the field has evolved in layers, each trying to fix the limitations of the previous one[4].

A. *Rainfall as a Trigger*

The earliest and most intuitive line of work focused on rainfall thresholds. The idea was simple: if rainfall intensity and duration cross a certain limit, a landslide is likely to occur. This approach goes back to classic studies like Caine's work, which established global intensity-duration relationships.

But here's the catch—these thresholds don't behave consistently everywhere. A rainfall event that triggers a landslide in one region might do nothing in another. Over time, researchers realized that it's not just *how much* it rains, but also *what happened before*. Soil that has already been soaking for days reacts very differently compared to dry ground[7].

That's where the concept of antecedent rainfall came in. Studies began incorporating rainfall accumulated over several days or even weeks, and the improvement in prediction accuracy was noticeable. It makes sense in a physical way too—landslides are not instant reactions; they are the result of gradual weakening.

B. *Machine Learning Approaches*

As datasets grew larger, machine learning started to take over. Models like Random Forest and Support Vector Machines became popular because they could handle complex, nonlinear relationships between variables like rainfall, slope, and vegetation

Random Forest, in particular, gained attention for being robust and relatively easy to interpret. In several regional studies, it achieved strong performance metrics, often outperforming simpler statistical methods. However, most of these models still relied on static input features. They treated each data point as independent, without considering how conditions evolved over time.

That limitation becomes obvious when you think about rainfall again—it's not a single value; it's a sequence[9].

C. *Deep Learning and Temporal Modelling*

This is where deep learning, especially LSTM networks, started making a real difference. Unlike traditional models, LSTMs are designed to handle sequences. They can remember patterns over time, which makes them well-suited for rainfall data.

Some studies showed that even adding just a few days of prior rainfall data significantly improved predictions. Others went further, using longer sequences and more complex architectures. When CNNs were combined with LSTMs, the models could capture both spatial features (like terrain) and temporal patterns (like rainfall trends)[12].

There's something interesting here though—while these models perform well, they also become harder to interpret. You get better accuracy, but less transparency.

D. *Regional Adaptation and Transfer Learning*

One issue that keeps coming up across studies is generalization. A model trained in one region rarely performs equally well in another. This isn't surprising, given how much environmental conditions vary across locations.

To address this, researchers have started exploring region-specific modeling and clustering techniques. Instead of treating an entire country as one unit, they divide it into smaller zones with similar characteristics. This improves local accuracy but doesn't fully solve the problem of scalability.

Transfer learning offers a more flexible approach. The idea is to train a model on a large dataset from one region and then adapt it to another region with limited data. While this technique is widely used in fields like computer vision, its application in landslide prediction is still relatively new[4].

Recent work suggests that combining transfer learning with deep learning architectures can significantly improve performance in data-scarce regions. Still, this area is far from mature, and there is plenty of room for improvement.

E. Summary of Existing Approaches

If you step back and look at the bigger picture, a pattern emerges:

- Threshold models are simple but lack flexibility
- Machine learning models improve accuracy but ignore temporal dynamics
- Deep learning models capture temporal behavior but struggle with generalization
- Regional models improve local performance but reduce scalability

This progression naturally leads to the idea behind our work—combining temporal modeling with regional adaptability in a single unified framework[7].

The comparison of these approaches is summarized in Table II.

TABLE II: Comparison of Existing Landslide Prediction Approaches

Approach	Strength	Limitation
Threshold Models	Simple, easy to implement	Poor generalization
Machine Learning	Captures nonlinear patterns	No temporal modeling
Deep Learning	Handles time-series data	Low interpretability
Regional Models	High local accuracy	Not scalable

This gap—between accuracy, adaptability, and scalability—is exactly what the proposed Region-Adaptive AI Framework (RAAF) aims to address.

III. Research Gap Problem Statement

Even with all the progress in landslide prediction over the years, something still feels. . . incomplete. You see models reporting high accuracy, impressive metrics, and yet, when applied in a different region or under slightly different conditions, their performance drops noticeably.

If we look closely, the limitations are not hard to spot.

First, traditional rainfall-threshold models are simply too rigid. They rely on predefined relationships between rainfall intensity and duration, which might work in one region but fail in another. The natural world isn't that consistent—soil type, vegetation, slope structure, and even human activity can change how a slope behaves under rainfall[5].

Machine learning improved things by learning patterns directly from data. Models like Random Forest and SVM can capture complex relationships between environmental variables. But most of these approaches still treat the data as static. They don't fully capture how rainfall evolves over time, which, in reality, is one of the most critical aspects of landslide formation.

Deep learning models, especially LSTM-based approaches, took a step forward by introducing temporal modeling. They can learn from sequences and identify patterns across multiple time steps. However, many existing studies use limited temporal windows—often just a few days—which may not be sufficient to capture long-term soil saturation effects. In regions where landslides are triggered by prolonged rainfall rather than sudden storms, this becomes a serious limitation.

Then comes perhaps the biggest challenge: generalization.

Most models are trained on data from a single region. They learn patterns specific to that environment—its rainfall behavior, terrain, and geology. When the same model is applied elsewhere, those learned patterns don't always hold. It's a bit like training for mountain driving in Himachal and then being asked to drive in the Western Ghats—the conditions are similar, but not identical[3].

Another practical issue is data availability. Many high-risk regions simply don't have enough recorded landslide events to train complex models. Deep learning methods, in particular, tend to struggle in such data-scarce environments. Without enough examples, the model either overfits or fails to learn meaningful patterns.

There's also a structural gap in how models are designed. Spatial features (like terrain and vegetation) and temporal features (like rainfall over time) are often handled separately[6]. Few approaches successfully integrate both in a way that reflects how landslides actually occur—as a combination of location-specific conditions and time-dependent triggers.

Putting all of this together, the problem becomes clearer.

There is a need for a system that can:

- Understand rainfall not as a single value, but as a sequence evolving over time
- Incorporate spatial terrain characteristics alongside temporal patterns
- Adapt to different geographic regions without requiring large amounts of new data
- Maintain reliable performance even in data-scarce conditions

This research addresses these challenges by proposing a Region-Adaptive Artificial Intelligence Framework (RAAF). The framework combines multi-temporal rainfall modeling, spatial feature extraction using CNN, and a transfer learning mechanism that allows knowledge to be reused across regions.

In simple terms, the goal is not just to build a more accurate model—but to build one that can actually be used in the real world[1], where data is messy, regions differ, and perfect conditions rarely exist.

IV. Methodology

Instead of treating landslide prediction as a single-step problem, the proposed framework approaches it as a pipeline—where each stage adds a layer of understanding. The idea is to gradually move from raw environmental data to a meaningful risk prediction, while preserving both spatial and temporal context.

At a high level, the Region-Adaptive Artificial Intelligence Framework (RAAF) is organized into four main stages: data collection, feature engineering, deep learning-based prediction, and region-adaptive learning. The overall flow of the system is illustrated earlier in Fig. 1, and each stage is described below.

A. Data Collection

The first step is gathering data from multiple sources. This includes rainfall data, terrain information, vegetation indices, soil moisture levels, and historical landslide records.

Rainfall data is obtained from sources like IMD and satellite-based datasets such as NASA GPM, which are particularly useful in regions where ground stations are sparse. Terrain-related features like slope and elevation are derived from Digital Elevation Models (DEM), while vegetation and soil-related information comes from satellite products such as MODIS and SMAP.

One thing worth noting here is that no single dataset is complete on its own. The strength of the framework comes from combining these sources to form a more complete picture of the environment.

B. Preprocessing and Feature Engineering

Raw data is rarely ready for direct use. There are missing values, inconsistencies, and different scales across datasets. So, preprocessing becomes an essential step.

Missing values are handled through interpolation or imputation, and all features are normalized to ensure consistency during model training. After that, the focus shifts to feature engineering—particularly the construction of multi-temporal rainfall features.

Instead of using just current rainfall, the model considers rainfall accumulated over different time windows:

- Short-term rainfall (last 3 days)
- Medium-term rainfall (last 7 days)
- Long-term rainfall (last 30 days)

This multi-scale representation allows the model to capture both immediate triggers and gradual soil saturation effects.

In some cases, techniques like Variational Mode Decomposition (VMD) can be applied to separate rainfall signals into different frequency components, helping the model focus on meaningful patterns rather than noise.

C. CNN-LSTM Prediction Model

At the core of the framework lies a hybrid deep learning model that combines CNN and LSTM.

The CNN component is responsible for extracting patterns from spatial features. Think of it as identifying terrain-related signals—like how slope or vegetation might influence landslide susceptibility.

The LSTM component, on the other hand, handles temporal sequences. It processes rainfall data over time, learning how past conditions influence current risk. Unlike traditional models, it doesn't treat each time step independently—it builds a memory of past rainfall.

To improve performance further, an attention mechanism is introduced. This allows the model to assign more importance to specific time periods—for example, focusing more on the days when rainfall was particularly intense or sustained.

Together, this combination allows the model to understand not just *where* a landslide might occur, but also *when* conditions become critical.

D. Region-Adaptive Learning

This is where the framework becomes a bit more practical for real-world use.

Instead of assuming that one model will work everywhere, the study area is divided into smaller regions based on geo-environmental similarity. These regions share similar characteristics like slope patterns, rainfall behavior, and vegetation cover.

For regions with sufficient historical data, models are trained normally. But for regions where data is limited—a very common situation—transfer learning is used.

In this process, a model trained on a data-rich region is reused. The lower layers of the model, which capture general patterns, are kept unchanged, while the upper layers are fine-tuned using whatever limited data is available in the target region.

This approach significantly reduces the need for large datasets while still maintaining reasonable prediction accuracy.

E. Risk Classification

Finally, the model outputs a probability score indicating the likelihood of a landslide. To make this more interpretable, the probabilities are divided into four categories:

- Low Risk (less than 25
- Moderate Risk (25
- High Risk (50
- Critical Risk (above 75

These categories are easier to understand and can be directly used in early warning systems or decision-making processes.

F. Overall Pipeline Summary

To summarize, the methodology follows a structured pipeline, as shown in Table III.

TABLE III: RAAF Pipeline Summary

Stage	Description
Data Collection	Multi-source environmental data
Preprocessing	Cleaning and normalization
Feature Engineering	Multi-temporal rainfall features
CNN-LSTM Model	Spatio-temporal prediction
Region Adaptation	Transfer learning
Risk Output	Probability classification

If you look at it as a whole, the framework tries to mirror how landslides actually occur—through a combination of spatial conditions and time-dependent triggers—rather than treating them as isolated events.

V. System Architecture

Instead of treating landslide prediction as a single-step problem, the proposed framework approaches it as a pipeline—where each stage adds a layer of understanding. The idea is to gradually move from raw environmental data to a meaningful risk prediction, while preserving both spatial and temporal context.

At a high level, the Region-Adaptive Artificial Intelligence Framework (RAAF) is organized into four main stages: data collection, feature engineering, deep learning-based prediction, and region-adaptive learning. The overall flow of the system is illustrated earlier in Fig. 1, and each stage is described below.

A. Data Collection

The first step is gathering data from multiple sources. This includes rainfall data, terrain information, vegetation indices, soil moisture levels, and historical landslide records.

Rainfall data is obtained from sources like IMD and satellite-based datasets such as NASA GPM, which are particularly useful in regions where ground stations are sparse. Terrain-related features like slope and elevation are derived from Digital Elevation Models (DEM)[3], while vegetation and soil-related information comes from satellite products such as MODIS and SMAP.

One thing worth noting here is that no single dataset is complete on its own. The strength of the framework comes from combining these sources to form a more complete picture of the environment.

B. Preprocessing and Feature Engineering

Raw data is rarely ready for direct use. There are missing values, inconsistencies, and different scales across datasets. So, preprocessing becomes an essential step.

Missing values are handled through interpolation or imputation, and all features are normalized to ensure consistency during model training[13]. After that, the focus shifts to feature engineering—particularly the construction of multi-temporal rainfall features.

Instead of using just current rainfall, the model considers rainfall accumulated over different time windows:

- Short-term rainfall (last 3 days)
- Medium-term rainfall (last 7 days)
- Long-term rainfall (last 30 days)

This multi-scale representation allows the model to capture both immediate triggers and gradual soil saturation effects.

In some cases, techniques like Variational Mode Decomposition (VMD) can be applied to separate rainfall signals into different frequency components, helping the model focus on meaningful patterns rather than noise.

C. CNN-LSTM Prediction Model

At the core of the framework lies a hybrid deep learning model that combines CNN and LSTM.

The CNN component is responsible for extracting patterns from spatial features. Think of it as identifying terrain-related signals—like how slope or vegetation might influence landslide susceptibility[11].

The LSTM component, on the other hand, handles temporal sequences. It processes rainfall data over time, learning how past conditions influence current risk. Unlike traditional models, it doesn't treat each time step independently—it builds a memory of past rainfall.

To improve performance further, an attention mechanism is introduced. This allows the model to assign more importance to specific time periods—for example, focusing more on the days when rainfall was particularly intense or sustained.

Together, this combination allows the model to understand not just *where* a landslide might occur, but also *when* conditions become critical.

D. Region-Adaptive Learning

This is where the framework becomes a bit more practical for real-world use.

Instead of assuming that one model will work everywhere, the study area is divided into smaller regions based on geo-environmental similarity. These regions share similar characteristics like slope patterns, rainfall behavior, and vegetation cover.

For regions with sufficient historical data, models are trained normally. But for regions where data is limited—a very common situation—transfer learning is used[1].

In this process, a model trained on a data-rich region is reused. The lower layers of the model, which capture general patterns, are kept unchanged, while the upper layers are fine-tuned using whatever limited data is available in the target region.

This approach significantly reduces the need for large datasets while still maintaining reasonable prediction accuracy.

E. Risk Classification

Finally, the model outputs a probability score indicating the likelihood of a landslide. To make this more interpretable, the probabilities are divided into four categories:

- Low Risk (less than 25
- Moderate Risk (25
- High Risk (50
- Critical Risk (above 75

These categories are easier to understand and can be directly used in early warning systems or decision-making processes.

F. Overall Pipeline Summary

To summarize, the methodology follows a structured pipeline, as shown in Table III.

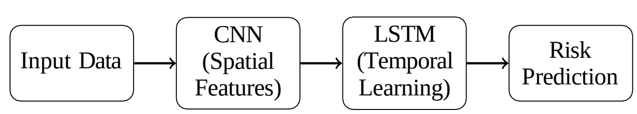

Fig. 2: CNN-LSTM based system architecture

If you look at it as a whole, the framework tries to mirror how landslides actually occur—through a combination of spatial conditions and time-dependent triggers—rather than treating them as isolated events.

VI. Challenges And Limitations

No matter how well a model performs on paper, real-world conditions tend to expose its weaknesses. The proposed framework is no exception[9]. While it improves prediction accuracy and adaptability, there are still several practical challenges that need to be considered.

A. Data Availability and Quality

One of the biggest constraints in landslide prediction is the availability of reliable data. In many high-risk regions, especially remote mountainous areas, landslide events are not consistently recorded. Smaller events often go undocumented, and datasets tend to be biased toward major incidents.

This creates an imbalance during model training. The model ends up learning from incomplete or skewed data, which can affect its ability to generalize. Even though the framework uses transfer learning to handle data-scarce regions[7], the quality of predictions still depends heavily on how representative the available data is.

Rainfall data presents its own set of challenges. In mountainous terrain, rainfall can vary significantly over short distances due to orographic effects. Ground-based rain gauges may not capture this variation accurately, and satellite data, while useful, may introduce its own uncertainties.

B. Model Interpretability

Deep learning models are often criticized for being difficult to interpret, and this framework is no different. While the CNN-LSTM architecture is effective at capturing complex patterns, it doesn't always provide clear explanations for its predictions.

This becomes a serious concern in disaster management scenarios. A false alarm can lead to unnecessary evacuations, while a missed prediction can have severe consequences. Decision-makers usually prefer systems that not only predict outcomes but also explain *why* those predictions are made.

The attention mechanism helps to some extent by highlighting which time periods contributed most to a prediction. However, the internal workings of the CNN and LSTM layers remain largely opaque. Incorporating explainable AI techniques could make the system more trustworthy in practical applications[11].

C. Non-Stationarity and Climate Variability

Another challenge that is becoming increasingly important is the changing nature of climate patterns. The relationship between rainfall and landslides is not fixed—it evolves over time.

Models trained on historical data assume that future conditions will behave similarly. But with changing rainfall intensity, shifting seasons, and extreme weather events becoming more frequent, this assumption may not always hold.

The framework attempts to address short-term variability by using multi-temporal rainfall features, but it does not fully account for long-term climate shifts. Periodic retraining with updated data, or incorporating climate projections, may be necessary to maintain accuracy over time.

D. Real-Time Implementation Constraints

While the model performs well using historical datasets, deploying it in real-time systems introduces additional challenges. Real-time prediction requires continuous data streams, reliable infrastructure, and fast processing capabilities[3].

Integrating the framework with live rainfall data, IoT sensors, or satellite feeds would be essential for practical early warning systems. However, such integration involves technical and logistical challenges that go beyond model design.

E. Overall Perspective

Taken together, these limitations highlight an important point: building an accurate model is only part of the problem. Making that model reliable, interpretable, and usable in real-world conditions is an entirely different challenge.

The proposed framework moves in that direction, but there is still room for improvement—especially in terms of data quality[13], explainability, and real-time deployment.

VII. Experimental Setup

Once the framework was designed, the next step was to test whether it actually works in practice—and not just under ideal conditions. The experimental setup was structured to reflect realistic scenarios, including variations in data sources, feature types, and model configurations.

A. Dataset Description

The study uses a combination of datasets from both Indian and global sources. Rainfall data was collected from the India Meteorological Department (IMD) and complemented with satellite-based observations from NASA's Global Precipitation Measurement (GPM) mission[6]. This combination helps balance accuracy and coverage, especially in areas where ground stations are sparse.

For terrain-related features, Digital Elevation Model (DEM) data from SRTM was used to derive slope and elevation. Vegetation information was obtained using the NDVI index from MODIS datasets, while soil moisture data was sourced from SMAP. Historical landslide records were taken from the ISRO Landslide Atlas of India and the NASA Global Landslide Catalog.

Bringing these datasets together wasn't entirely straightforward. They differ in resolution, time frequency, and format, so a fair amount of preprocessing was required to align them before model training.

B. Feature Variables

To make the model more informative, both temporal and spatial features were included. Rainfall was not treated as a single value but as a sequence capturing different time windows. Alongside this, terrain and environmental features were added to provide context.

The key input variables used in the model are summarized in Table IV.

What's important here is not just the number of features, but how they complement each other. Rainfall captures the trigger[5], while slope, vegetation, and soil moisture describe how the land responds to that trigger.

TABLE IV: Input Feature Variables

Feature	Description
Rainfall (24h)	Instantaneous rainfall (mm)
Rainfall (7-day)	Weekly accumulated rainfall
Rainfall (30-day)	Monthly accumulated rainfall
Slope Gradient	Derived from DEM
Soil Moisture	Saturation level (%)
NDVI	Vegetation index

C. Model Configuration

The CNN-LSTM model was configured with a structure that balances complexity and efficiency. The CNN component consists of three convolutional layers with a kernel size of 3×3, followed by ReLU activation and max-pooling layers to reduce dimensionality.

The LSTM component includes 128 units and is designed to process rainfall sequences over a 30-day window. A dropout rate of 0.2 is applied to reduce overfitting, especially given the variability in environmental data.

For training, the Adam optimizer was used with a learning rate of 0.001, and the loss function was categorical cross-entropy. The model was trained for up to 100 epochs, with early stopping applied based on validation performance to prevent unnecessary over-training.

D. Training Strategy

Instead of randomly splitting the data, a temporal train-test split was used—typically 80

Performance was evaluated using standard metrics such as accuracy, F1-score, and Area Under the Curve (AUC). These metrics provide a balanced view of the model's performance, especially in cases where class distribution may be uneven.

E. Overall Setup Perspective

If you look at the setup as a whole, the intention was not just to achieve high accuracy, but to simulate realistic conditions[10]. The use of multi-source data, temporal splitting, and multiple evaluation metrics ensures that the results are meaningful and not just optimized for a controlled environment.

VIII. Results

After training and testing the proposed framework, the results offer a fairly clear picture—incorporating both temporal rainfall patterns and regional adaptability does make a noticeable difference.

The CNN-LSTM model was evaluated against commonly used baseline models, including Random Forest, Support Vector Machines (SVM), and Artificial Neural Networks (ANN). These models were chosen because they are widely used in landslide prediction studies and provide a meaningful benchmark.

The performance comparison is summarized in Table V.

At first glance, the improvement might look like just a few percentage points. But in the context of disaster prediction[8], even a small increase in accuracy can translate into significantly better risk assessment and decision-making.

TABLE V: Model Performance Comparison

Model	Accuracy	F1 Score	AUC
Random Forest	83.2%	0.81	0.86
SVM	81.5%	0.79	0.84
ANN	84.7%	0.83	0.88
CNN-LSTM	**91.1%**	**0.89**	**0.93**

To visualize the comparison more clearly, Fig. 4 presents a graphical representation of model performance.

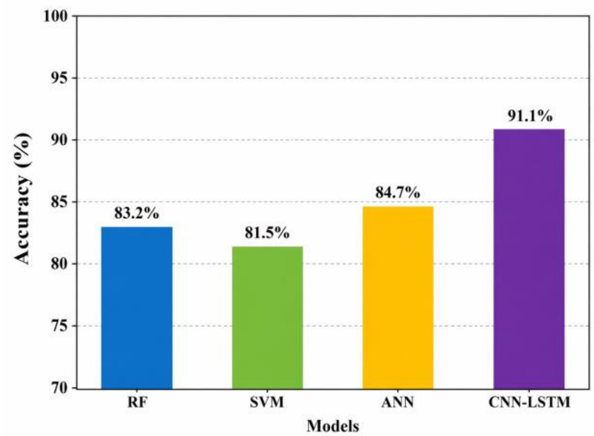

Fig. 3: Performance Comparison of Different Models

Looking a bit deeper into the results, a few patterns stand out.

First, the CNN-LSTM model consistently performs better across all evaluation metrics. This suggests that combining spatial and temporal modeling provides a more complete understanding of landslide behavior compared to models that rely on static inputs.

Second, the inclusion of multi-temporal rainfall features appears to play a significant role. Models that consider rainfall over longer periods—such as 7-day and 30-day windows—are better at capturing gradual soil saturation effects, which are often critical in landslide formation.

Third, the region-adaptive mechanism contributes to improved generalization. During testing, the model maintained stable performance[1] even when applied to regions with limited training data, indicating that transfer learning helped retain useful patterns from data-rich regions.

However, it's also important to keep expectations realistic. The model does not eliminate prediction errors entirely. There are still cases where landslides occur under conditions that are not strongly represented in the training data, or where environmental factors interact in unexpected ways.[3]

Overall, the results suggest that the proposed framework offers a meaningful improvement over traditional approaches—not just in terms of accuracy, but also in its ability to adapt across different regions and conditions.

IX. Discussion

Looking at the results as a whole, a few things become quite clear—some expected, some a bit more revealing.

To begin with, the importance of multi-temporal rainfall stands out strongly. The model performs noticeably better when it considers rainfall accumulated over several days rather than relying on a single-day value. In fact, the 7-day and 30-day rainfall windows consistently emerge as strong indicators of landslide risk. This aligns well with what field studies have suggested for years—that landslides are often the result of gradual soil saturation rather than just sudden heavy rainfall.[1]

The LSTM component plays a key role here. By processing rainfall as a sequence rather than isolated values, it captures patterns that simpler models tend to miss[2]. It's almost like the model develops a "memory" of how conditions have been evolving, which turns out to be quite important for prediction[9].

One of the more interesting observations comes from the region-adaptive mechanism. The ability to transfer knowledge from one region to another helps maintain performance even when local data is limited. This is particularly useful in areas where landslide records are sparse, which is often the case in remote or developing regions[9].

That said, the improvement is not just about accuracy—it's about consistency. A model that performs well only in one region is of limited use. The fact that this framework maintains stable performance across different regions suggests that the combination of transfer learning and geo-environmental grouping is working as intended.

However, there are still some limitations worth noting. The model relies entirely on historical data, which means it reflects past patterns. If environmental conditions change significantly—due to climate shifts or land-use changes—the model may need retraining to stay relevant.

Another point is the lack of real-time integration. While the framework is capable of making accurate predictions, it is not yet connected to live data streams. In practical scenarios, this would be a necessary step for deployment in early warning systems.

Overall, the discussion reinforces a key idea: landslide prediction is not just about choosing the most advanced model, but about combining the right types of information—temporal, spatial, and regional—into a single[3], coherent system.

And in that sense, the proposed framework seems to move in the right direction.

X. Conclusion

This study set out to address a fairly practical problem—how to make landslide prediction models not only accurate, but also adaptable across different regions. As it turns out, achieving both at the same time is not straightforward.

The proposed Region-Adaptive Artificial Intelligence Framework (RAAF) brings together three key ideas: modelling rainfall over multiple time scales, capturing spatial terrain characteristics, and enabling cross-regional adaptability through transfer learning. Instead of treating landslides as isolated events, the framework approaches them as outcomes of evolving environmental conditions.

The results show that incorporating multi-temporal rainfall features significantly improves prediction performance. In particular, rainfall accumulated over 7-day and 30-day periods plays an important role, reinforcing the idea that landslides are often driven by gradual soil saturation rather than just short-term intensity spikes.

The hybrid CNN-LSTM architecture proved effective in capturing both spatial and temporal patterns. More importantly, the region-adaptive component allowed the model to maintain stable performance even in areas with limited data, which is a common challenge in real-world applications.

At the same time, the study also highlights certain limitations. The model relies on historical datasets and does not yet operate in real-time environments. Additionally, changes in climate patterns may affect long-term prediction reliability, making periodic updates necessary.

Overall, the framework represents a step toward more practical and scalable landslide prediction systems. It does not claim to solve the problem entirely—but it does move closer to a solution that can work beyond controlled experimental settings.

In the broader context, as extreme weather events become more frequent, the need for reliable early warning systems will only grow. Approaches that combine data-driven learning with adaptability, like the one proposed here, are likely to play an important role in that future.

References

[1] F. Guzzetti, M. Melillo, and A. C. Mondini, "Landslide predictions through combined rainfall threshold models," *Landslides*, vol. 22, pp. 137–147, 2025.

[2] J. Kang, B. Wan, and Z. Gao, "Machine learning forecasting for rainfall-induced landslides in Yunnan Province," *Scientific Reports*, vol. 14, p. 14049, 2024.

[3] Y. Liu, S. Ma, and L. Dong, "Comparative study of regional rainfall-induced landslide early warning models based on RF, CNN, and MLP," *Frontiers in Earth Science*, vol. 12, 2024.

[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[5] Nature Communications, "Deep learning forecast of rainfall-induced shallow landslides," 2023. doi: 10.1038/s41467-023-38135-y.

[6] H. Wang, Y. Ao, and C. Wang, "Dynamic prediction of landslide displacement using VMD-SSO-LSTM," *Scientific Reports*, vol. 14, p. 9203, 2024.

[7] L. Piciullo, M. Calvello, and J. M. Cepeda, "Territorial early warning systems for rainfall-induced landslides," *Earth-Science Reviews*, vol. 179, pp. 228–247, 2018.

[8] A. Dikshit, R. Sarkar, and B. Pradhan, "Rainfall-induced landslide studies in the Indian Himalayan region: A review," *Applied Sciences*, vol. 10, no. 7, p. 2466, 2020.

[9] N. Caine, "The rainfall intensity-duration control of shallow landslides," *Geografiska Annaler*, vol. 62, pp. 23–27, 1980.

[10] Y. Zongji, L. Wang, and J. Qian, "Multivariate real-time early warning method for rainfall-induced landslides," *Landslides*, vol. 17, no. 10, pp. 2409–2419, 2020.

[11] Springer, "Regional dynamic hazard assessment guided by geographic similarity," *Bulletin of Engineering Geology*, 2024.

[12] ISRO NRSC, "Landslide Atlas of India," Hyderabad, India, 2021.

[13] NASA, "Global Precipitation Measurement (GPM) Mission Data," NASA GSFC, 2023.