

Decentralized Landslide Disaster Prediction for Imbalanced and Distributed Data

Ren Ozeki

Osaka University, Japan
Japan Meteorological Agency
r-ozeki@ist.osaka-u.ac.jp

Haruki Yonekura

Osaka University
Osaka, Japan
h-yonekura@ist.osaka-u.ac.jp

Hamada Rizk

Osaka University, Osaka, Japan
Tanta University, Tanta, Egypt
hamada_rizk@ist.osaka-u.ac.jp

Hirozumi Yamaguchi

Osaka University
Osaka, Japan
h-yamagu@ist.osaka-u.ac.jp

Abstract—Precise disaster prediction plays a critical role in saving lives. Traditional landslide prediction methods have predominantly relied on deep neural networks such as CNNs and LSTMs. However, these methods face two main challenges. The first is the class imbalance issue, as landslides are rare, disrupting the training process. The second challenge stems from decentralized data management, with variations in volume and characteristics across regions. Typically, local municipalities manage disaster data within their regions, and sharing or migrating this data is not straightforward. This paper presents *SlideSafe*: a novel landslide prediction system that combines spatio-temporal contrastive learning and collaborative learning¹. It begins by training a contrastive learning model to extract meaningful representations of land characteristics in each region. Subsequently, these trained models are merged among regions with similar characteristics, leveraging federated learning. The federated models are then fine-tuned and customized for the landslide event prediction using the data specific to each region. Experimental results indicate that the proposed system achieves higher precision under 100% recall compared to state-of-the-art federated learning methods, which are often adversely affected by non-iid data and data scarcity.

Index Terms—Decentralized Landslide Prediction, Class imbalance, Contrastive Learning, Selective Federated Learning

I. INTRODUCTION

Accurate disaster prediction is essential for safeguarding lives through disseminating warnings, providing vital time for evacuation, and directing people to secure locations. Global climate change has heightened the need for improving global disaster management systems. Specifically, rainfall-induced landslides are particularly vulnerable to climate change, potentially occurring in historically safe areas. Unlike other disasters, predicting landslides is challenging due to complex factors like geological conditions, terrain, rainfall patterns, and soil composition, requiring urgent attention [1].

Traditionally, meteorological agencies worldwide have relied on statistical data processing from numerical weather models and observation systems for landslide prediction [2]. These traditional approaches frequently encounter challenges due to their incapacity to adequately account for the intricate spatial interdependencies and the evolving dynamics of water accumulation—both of which play crucial roles in landslide occurrences. In light of these constraints, existing methods

have turned to harnessing the power of deep neural networks, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks [3], [4].

Predicting landslides practically presents a substantial challenge due to the scarcity of labeled data and class imbalance in datasets. Landslide events are rare, leading to an unequal distribution of positive and negative cases. For instance, in a dataset of weather-related incidents, there might be thousands of records of non-landslide events but only a handful of landslide occurrences. This class imbalance greatly hinders the training of machine/deep learning models, often resulting in a biased model that does not consider the minority class, which in this case, represents landslides. Due to the problems of class imbalance and the scarcity of labeled data, current methods do not attain nearly 100% recall while maintaining a reasonably low false positive rate, a critical requirement for enabling residents to respond effectively to disasters [5].

Another challenge arises due to the decentralized nature of data management and data disparities among different regions. Federated learning has emerged as a promising solution for model training over distributed data. This can be attributed to the fact that landslides and other disasters are typically monitored and managed by local municipalities, which means that the relevant disaster data is stored with these municipal authorities. To be more specific, local municipalities require region-specific data and have established their own monitoring facilities. However, the data collection processes employed by local municipalities may not be consistent, resulting in variations in data quality and quantity. Besides, terrain characteristics and weather conditions vary significantly from one region to another. These heterogeneities lead to a non-independent and non-identically distributed (non-IID) data problem, which has been demonstrated to impact the efficiency of the training process of federated learning in a recent study [6], [7].

To address these challenges, we propose *SlideSafe*: a novel decentralized landslide prediction system. *SlideSafe* involves the use of contrastive learning within each municipality to capture spatio-temporal dependencies and patterns from imbalanced and distributed data. Contrastive learning has garnered significant attention in recent years for its ability to effectively learn high-quality representations from complex input data, irrespective of label information. Subsequently, we quantify the similarity between each pair of municipalities based on

¹Our implementation is available here (<https://github.com/mclab-osaka/slidesafe>)

their encoded features. This similarity measurement guides a decentralized collaborative learning process where municipalities with similar features collaborate to construct a shared model (an encoder) to enhance the learning process. Finally, we fine-tune the shared model to adapt it to each municipality for accurate landslide prediction.

The proposed system underwent a thorough evaluation using real-world data from government agencies in Japan. The data was collected over two years to assess the system's effectiveness in precision under 100% recall. The obtained results validate the system's capability to achieve a remarkable landslide prediction performance of 62.4%. This represents a substantial improvement over the state-of-the-art techniques, surpassing them by up to 40.4%. These results demonstrate the feasibility of predictive accuracy in real-world applications.

II. RELATED WORK

A. Landslide prediction

Traditional landslide prediction approaches are algorithm-based methods [2], [8]. For instance, the author of [2] proposed a statistical method based on rail fall amount to issue early warning of landslides and evaluate their method using a dataset of India as a case study. However, since these works don't consider the spatio-temporal dependency of landslides, like soil distribution, elevation, and rainfall-related water accumulation, these methods have significant limitations in accuracy and generality regarding regions.

To capture spatio-temporal dependency of landslides, machine learning methods are also proposed [3], [4]. To predict landslides accurately by capturing spatial dependency, the author of [3] tries to predict landslide events with CNN. The literature [4] tries to predict the displacement of the ground that is a sign of landslide by utilizing LSTM.

Although these existing approaches are promising in terms of accuracy, to operate a landslide prediction system officially, it is more important to keep the 100% recall to avoid a situation where residents can't evacuate. simultaneously disaster prediction systems need to keep high precision because people do not trust low-precision information. Since landslides are unusual and rare events, it is challenging to achieve high precision while keeping 100% recall in landslide prediction due to class imbalance. However, these existing researches don't consider this class imbalance problem and the specific requirements of disaster systems. *Unlike existing research, SlideSafe tackles the class imbalance problem in landslide prediction and the particular requirements of disaster systems.*

B. Machine Learning from Imbalanced Data

The class imbalance problem is one of the fundamental problems of machine learning [9]. For instance, in binary classification, the presence of majority and minority classes is assumed, with a specific imbalance ratio. Since the standard classifier is optimized by a loss function that treats both classes equally, the minority class is almost ignored. However, the minority class is more important to detect in usual cases that

include our target research area. To overcome this class imbalance problem, many researches have addressed this problem [10]–[12]. Approaches to solving class imbalance problems are mainly categorized into two: Data-level approach and algorithm-level approach.

The data-level approach aims to obtain balanced training datasets from imbalanced dataset utilizing undersampling or oversampling [13]–[15]. Since undersampling methods risk removing data important for a model to learn the boundary, oversampling gains more attention than undersampling [14], [16]. To generate similar additional data even for multi-modal data, some data augmentation methods using generative models are proposed [16] because it is difficult for the traditional static approach such as SMOTE [14] to generate additional high-quality multi-modal data. The algorithm-level approach works directly during the training procedure of the classifier, disliking the previous data-level approach. The most commonly addressed issue with the algorithm-level approach is loss function adaptation, such as Focal loss [17], cost sensitive learning [18] and Mean False Error [19]. Mean False Error [19] balances the weight of loss from minority and majority classes. Focal loss [17] reduces the impact of easy instances on the loss function.

While the data-level approach holds promise, the cost of data augmentation can be substantial, especially in scenarios with high-class imbalance ratios or multi-modal input data. Conversely, relying solely on loss function adaptation often proves ineffective when the classifier is a deep, complex neural network because the amount of minority data is insufficient for obtaining high-quality latent representation. *To obtain meaningful latent representation, SlideSafe employs contrastive learning as self-supervised representation learning that is robust to label imbalance [20].*

C. Federated learning

Federated learning (FL) is widely used when it is difficult to collect distributed data on a server due to privacy, security, or data migration costs because FL enables distributed clients to train a shared model collaboratively without exchanging local data [21]. However, it is known that traditional FL can't train efficiently when global data distribution and local data distribution differ, which is called non-IID [22]. To build an effective model for global distribution in non-IID settings, many existing works address FL in non-IID settings [7], [22]–[24]. FedProx [22] is designed to avoid the local model deviating greatly from the global model.

Although these works succeed in converging learning and improving the global model under non-IID data, the author of [25]–[27] proposed personalization in FL because one global model cannot fit all clients. One example is FedProx-FT [25], which refines the global model from FedProx [22] using local data to create personalized models. One of the similarity-based approaches is FedAMP [26], which keeps the personalized cloud model for each client trying to learn in each party without sharing each client's data. MOON [7] and FedMoCo [24] use federated learning only between similar nodes whose

representations are similar. This approach usually does not lead to model convergence and thus affects the system performance. Additionally, none of the above-mentioned techniques has been applied to the field of disaster. *In contrast, SlideSafe adopts a decentralized collaborative learning mechanism and neighbor-finding way based on weight similarity that ensures model convergence and thus saves in communication costs as demonstrated in [28].*

III. PROBLEM STATEMENT

There are various types of land-related disasters, including landslides, mudslides, slope failures, surface/deep collapses, and debris flows resulting from heavy rainfall. In this paper, our primary focus is on *landslides triggered by rainfall*. Each local municipality, responsible for monitoring and administering landslide events, weather conditions, topographical features, and vegetation within its respective geographical area, shall henceforth be referred to as a "client". We define $\mathbf{X}_{a,b}^i$ and $y_{a,b}^i$ as the features and the label of client i during specific time period $[T_a, T_b]$, respectively. The notations \mathbf{X}^i and y^i do not specify time periods, and \mathbf{X}_t^i and y_t^i are the values in the specific moment t .

SlideSafe aims to predict landslide occurrence in the next time slot T_{t+1} where T_t is the current time. It takes into account two key input factors: time-series rainfall data and stable land characteristics that remain constant throughout the time period. The land characteristics encompass a range of features, including soil properties (categorized into 10 distinct classes), vegetation types (categorized into 11 classes), and elevation slope data (represented as real numbers with eight categorized directions). These land characteristics serve as an essential input for the system. Additionally, the system utilizes time series rainfall data, which consists of non-negative real numbers. This rainfall data is collected from government public agencies, ensuring the reliability and accuracy of the input information. As reported in a recent study [29], the latest numerical weather forecast is so accurate within a day that it can be considered actual value. Therefore, we can use the forecasted future rainfall amount data $X_{t+1,t+u}^i$ ($1 \leq u \leq \Lambda - t$) as well as the past observation $X_{t-h,t}^i$ ($0 \leq h \leq t-1$) as input for the prediction model. Our prediction task, aimed at client i for predicting the binary label y^i at time T_{t+1} , is formalized by Eq. (1). In Eq.(1), $\hat{y}_{t+1}^i \in \{0, 1\}$ represents the predicted binary outcome, indicating whether a landslide will occur at T_{t+1} . To achieve accurate landslide predictions for all clients' regions, *SlideSafe* minimizes the sum of all clients' loss ℓ , which is calculated using each client's model weight w_i , as expressed in Eq. (2).

$$\hat{y}_{t+1}^i = \mathcal{F}(\mathbf{X}_{t-h,t+u}^i) \quad (1)$$

$$\sum_{i \in [1, N]} \mathbb{E}_{z \sim D_i} [\ell(\mathbf{w}_i; \mathbf{X}^i)] \quad (2)$$

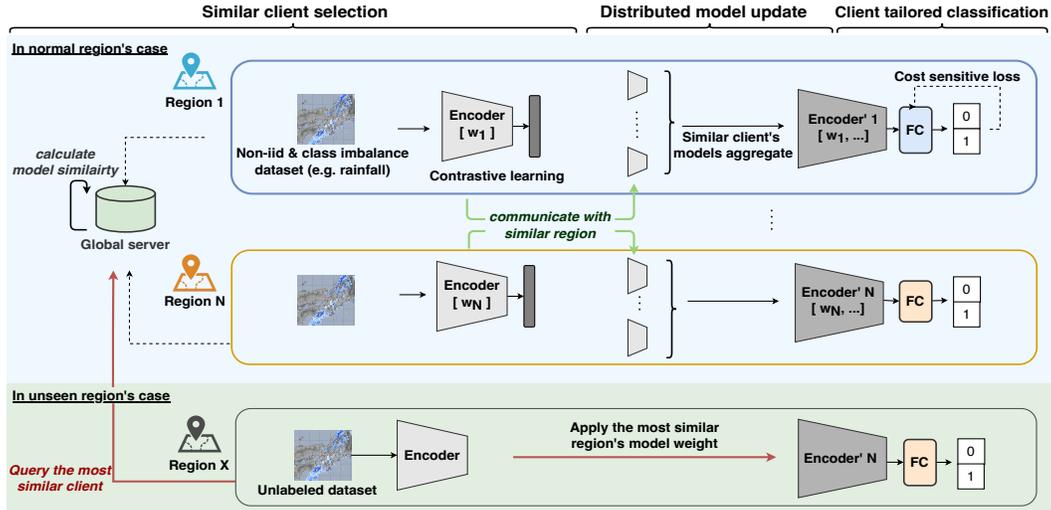
In this equation, $z = (\mathbf{X}^i, y^i)$ represents data following a probabilistic joint distribution D_i . Due to significant variations

in landslide occurrences and related features across regions, the data distribution D_i is heterogeneous and non-IID (non-independent and identically distributed).

This approach is driven by several key advantages of decentralized collaborative learning, including ensuring regional information security and facilitating comprehensive knowledge sharing across different regions. This knowledge-sharing aspect enables the system to adapt to any regional changes influenced by climate change or other factors. By adopting decentralized collaborative learning, *SlideSafe* leverages these advantages to create accurate, robust, and adaptable prediction models for diverse geographical areas. Furthermore, it is worth noting that D_i exhibits class imbalanced data distribution, which poses challenges for traditional machine learning and federated learning approaches in obtaining general knowledge or personalized models, especially within a decentralized collaborative learning framework. For instance, clients with no prior experience in landslides may participate in federated learning, potentially negatively impacting the learning process. *Despite the distinctiveness of each client's distribution D_i , SlideSafe employs decentralized personalized collaborative learning. In the proposed approach, the prediction model leverages knowledge from other clients to acquire comprehensive insights about landslides, enabling generalization to potential future changes in the region while also ensuring specialization for the target client and its associated distribution.*

IV. SlideSafe– LANDSLIDE PREDICTOR FOR DISTRIBUTED DATA

The objective of *SlideSafe* is to predict landslides using the model of each client in their respective regions. To preprocess raw data (such as the rainfall, soil information, and landslide occurrence points) into a manageable and interpretable format for machine learning models, the **virtual gridding module** converts all data into square grid cell formats. The formatted data is then input into the **feature extractor module** of each client to extract meaningful latent features from complex spatio-temporal input within their respective regions. This extraction is achieved using contrastive learning techniques, as described in [20], where training is conducted self-supervised. Contrastive learning is designed to reduce the distance between the representations of different augmented data of the same input (i.e., positive pairs), and increase the distance between the representations of augmented views of different input (i.e., negative pairs). Therefore, contrastive learning contributes to distinguishable latent representation even in a self-supervised manner. Then, clients communicate with each other to efficiently share their knowledge, which includes the weights of the contrastive learning models, through a **decentralized collaborative learning mechanism**. Within this mechanism, clients can identify and collaborate with other clients that have similar data distributions, enabling efficient collaborative learning. This mechanism identifies similar clients by measuring the similarity of the local model's update because the similar direction of model update helps convergence of

Fig. 1. *SlideSafe* overview in heterogenous environments

aggregated model and reflects similar data distribution as written in [28]. Finally, each client fine-tunes a shallow neural network, the **client-tailored classifier**, to predict landslides within their respective regions. This fine-tuning process utilizes the feature extractor trained in the previous module. We note that the virtual gridding module, feature extractor module, and client-tailored classifier are client-dependent functions that run on each client, while the decentralized, federated learning mechanism is run on a single global server. The architecture will be explained later in this section.

Since our focus is on a non-IID and class-imbalanced disaster dataset, *SlideSafe* needs to obtain comprehensive knowledge about landslides while personalizing the prediction model to predict landslides accurately in the target region. *SlideSafe* utilizes contrastive learning in the feature extraction phase to derive general knowledge about landslides, as it does not rely on label information. This enables us to obtain meaningful representation without feeding each component's class imbalance ratio, which sets it apart from existing methods for class-imbalanced federated learning [30]. Furthermore, *SlideSafe* can enhance the efficiency of the learning process for each client and personalize each model based on the similarity assessment of clients and corresponding grouping.

A. Virtual Gridding Module

To make natural phenomena that are continuously iterable by the machine learning model, this module is responsible for transforming real-world data into a square grid format for data discretization. *SlideSafe*'s default cell edge length is 1km, and all data is represented in this square-grid cell format.

B. Feature Extractor Module

1) *Multi-View Encoder*: Rainfall-induced landslide depends on rainfall and the land characteristics (*i.e.*, soil, plant, slope, and elevation). To capture the effects of these factors, we employ a multi-view encoder that receives the time series data (*i.e.*, rainfall) and the land characteristics. Since underground water flow plays a significant role in landslides, it is essential to consider the target area and its nearby surroundings to

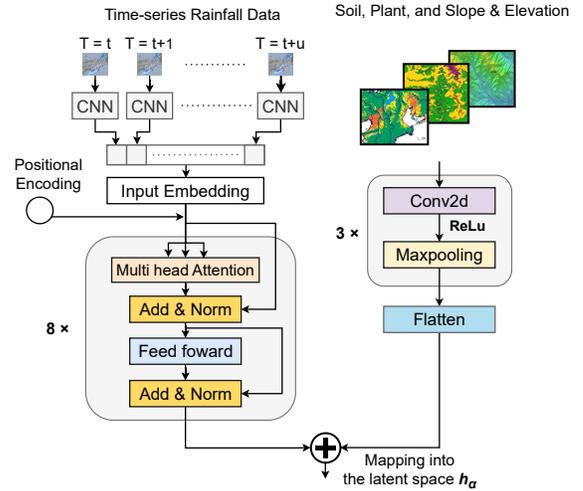


Fig. 2. Multi-input Feature Encoder. We employ eight transformer layers and three CNN layers.

accurately predict landslides. As shown in Figure 2, we extract the feature from a sequence of gridding data using ConvTransformer [31] and extract the feature from land characteristics data using CNN.

2) *Feature Extractor with Contrastive Learning*: This part obtains meaningful representation from the spatio-temporal inputs in grid-structured data. To avoid the effects of class imbalance and get feasible representation from complex spatio-temporal features, we incorporate a contrastive learning mechanism. Inspired by recent contrastive learning such as SimCLR [20] in the computer vision research domain, our feature extractor learns representations by maximizing agreement between differently augmented views of the same data example via a contrastive loss in the latent space. This framework comprises the following four major components. **Stochastic data augmentation**: From each data sample, it randomly generates two correlated views of the sample, denoted as \tilde{x}_α and \tilde{x}_β . They are considered a positive pair. In *SlideSafe*, we sequentially apply two simple augmentations: random cropping followed by resizing back to the original size, and

TABLE I
EXPERIMENTAL SETTINGS

Parameter	Explored range (Bold : default value)
Cell edge length [km]	{ 1 , 2}
Time interval [hour]	{ 6 , 12}
Batch size	{8, 16, 32, 64, 128 }
Temperature τ	{0.3, 0.5 , 0.8}
Initial communication round T_{init}	{10, 20, 50, 100 }
Communication round T	{10, 20, 30 }
Local epoch E	{10, 30 , 50}
Similarity threshold m	{-0.1, 0, 0.1, 0.3 , 0.6, 1.0}
Parameter of similarity metrics γ	{0.2, 0.4 , 0.6, 0.8}
Model size of <i>SlideSafe</i>	{0.76 Million}

random rotation to obtain \tilde{x}_α and \tilde{x}_β . **Encoder** $f(\cdot)$: It extracts representation vectors from the data augmented in the previous component. We employ an encoder that consists of two types of view to obtain $h_\alpha = f(\tilde{x}_\alpha)$ where $h_\alpha \in \mathbb{R}^d$ is a d -dimensional output of the multi-view encoder. **Small neural network projection head** $g(\cdot)$: It maps representations to the space where contrastive loss is applied. We use a shallow neural network with one hidden layer to obtain $z_\alpha = g(h_\alpha)$. **Contrastive loss prediction**: A contrastive loss function is defined for this contrastive prediction task, given a set $\{\tilde{x}_k\}$ including a positive pair \tilde{x}_α and \tilde{x}_β of samples.

We randomly pick up B examples for a minibatch and generate $2B$ augmented data using stochastic data augmentation. Then, we defined the contrastive prediction task on the augmented data, where for each positive pair, we applied contrastive learning regarding the rest of the $2(B-1)$ augmented examples as negative examples. Let $\text{sim}(u, v) = u^T v / \|u\| \cdot \|v\|$ denote the dot product between two normalized u and v (i.e. cosine similarity). Then the loss function for a positive pair of examples $(\tilde{x}_\alpha, \tilde{x}_\beta)$ is defined as:

$$\ell_{\alpha, \beta} = -\log \frac{\exp(\text{sim}(z_\alpha, z_\beta) / \tau)}{\sum_{k \in 2B_index_set} b_{[k \neq \alpha]} \exp(\text{sim}(z_k, z_\beta) / \tau)} \quad (3)$$

where the value of $b_{[k \neq \alpha]} \in \{0, 1\}$ is 1 if and only if $k \neq \alpha$, and τ denotes a temperature parameter. The final loss is computed across all positive pairs, both $(\tilde{x}_\alpha, \tilde{x}_\beta)$ and $(\tilde{x}_\beta, \tilde{x}_\alpha)$ in a mini-batch. This loss has been used in [20] and called *NT-Xent loss (the normalized temperature-scaled cross-entropy loss)*.

C. Decentralized Collaborate Learning Mechanism

This section presents a decentralized collaborative learning approach that ensures performance for each client. In general, model performance can be affected when aggregating models trained on dissimilar data distributions. To achieve high model performance even in non-IID settings, *SlideSafe* incorporates a similar neighbor selection mechanism [28]. This mechanism aims to identify clients whose data distribution is similar to each other, enhancing the performance of each model. It is called the personalized knowledge-sharing mechanism in *SlideSafe*, consisting of two stages: *similar client selection* and *distributed model update*.

1) *Similar Client Selection*: In this stage, the global server of *SlideSafe* aims to identify similar client for each local client (i.e. municipalities). Each local client is denoted as i and possesses a model with model parameters w_i . These parameters are updated over E local epochs and subsequently uploaded to a *global server* for a total of T_{init} rounds. This procedure mirrors the commonly used approach in centralized federated learning.

We measure the similarity of clients using the Personalized Adaptive Neighbor Matching (PANMGrad) [28] method, which does not require setting the number of clusters. The global server calculates the similarity of models of two clients i and j using a combination of two parameters. The first parameter is given in the first term of Eq. (4), where $\gamma \in [0, 1]$ is a hyperparameter, and g_i^t is the vectorized gradient of client i in round t and is expected to represent the data distribution in client i . g_i^t is obtained by $g_i^t = w_i^t - w_i^{t-1}$, where t is the current time round. We note that w_i^t is initialized by the same global model at the beginning of local training in each round, and one-round update g_i^t can be noisy. To capture the historical optimization directions of each model, we also introduce accumulated weight updates from the initial model, $h_i^t = w_i^t - w_i^0$, and use it in the second parameter given by the second term of Eq. (4). h_i^t is the accumulated vectorized gradient of client i in round t . The global server calculates the similarity of models of two clients using similarity metrics given in Eq. (4).

$$\text{sim}_{i,j} = \gamma \cdot \frac{\langle g_i^t, g_j^t \rangle}{\|g_i^t\| \cdot \|g_j^t\|} + (1 - \gamma) \cdot \frac{\langle h_i^t, h_j^t \rangle}{\|h_i^t\| \cdot \|h_j^t\|} \quad (4)$$

The global server selects the clients whose similarity is higher than the threshold m as the “neighbor clients” using average similarity during T_{init} . The threshold m is the system parameter.

2) *Distributed Model Update*: In this stage, each client communicates with each of the similar clients found in the previous stage. More concretely, the model parameters w_i of client i are delivered to each client j which is a similar client of i . Client i then waits for the models sent from up to N_{sim_i} clients and aggregates these received models. We use n to represent the number of received models, and it is important to note that $n \leq N_{\text{sim}_i}$, taking into account cases where model delivery may fail or not be performed due to some reasons. Eq. (5) is the federated aggregation of the received models.

$$\bar{w}_i^{t+1} \leftarrow \frac{1}{n+1} \sum_{j \in \{\text{similar clients of } i\} \cup \{i\}} w_j^t \quad (5)$$

The new aggregated model \bar{w}_i^{t+1} is then trained for E epochs before it is ready to be gossiped again.

D. Client Tailored Classification

This module takes the output (i.e., latent features) of the feature extractor module as input, and classifies the given inputs into positive and negative cases, which is output in confidence using the Sigmoid function. It is two layers shallow neural network and trained by a cost-sensitive loss function

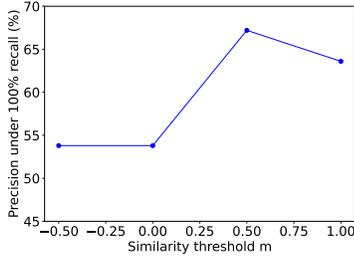
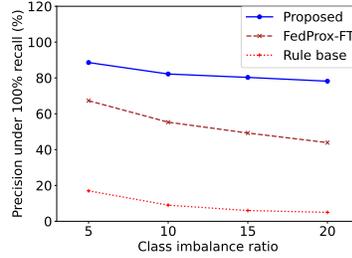
Fig. 3. Impact of Similarity Threshold m 

Fig. 4. Impact of Class Imbalance Ratio

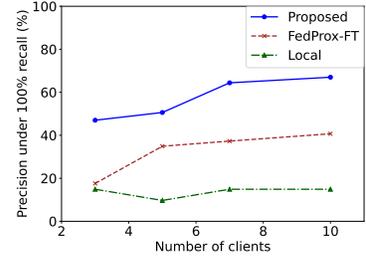


Fig. 5. Impact of Number of Clients

$\ell(y, \hat{y}) = -r \times y \log(\hat{y}) - (1 - y) \log(1 - \hat{y})$. The loss function calculates the loss according to the ratio of the classes, where y and \hat{y} are the actual label and output of the model respectively, and r is the ratio of the negative case and positive case.

Since landslide events are rare and geologically skewed, some clients may have never encountered landslides. In such situations, fine-tuning those clients may not work as they do not hold positive case data. To enable prediction for such clients (called *unlabeled clients*), *SlideSafe* provides the model among pre-trained models to those clients. Each unlabeled client, say u , can train its multi-view encoder and obtain model parameters w_u using $z_u \sim D_u$ through contrastive learning because contrastive learning does not rely on label information. Since the multi-view encoder is trained to obtain high-quality latent representation from rainfall and terrain data, the multi-view encoder's weight is expected to reflect rainfall and terrain patterns in this region. Thus, *SlideSafe* can select a similar client even for unlabeled clients. The global server selects the most similar model by comparing w_u with each model and finds the most similar client. Finally, unlabeled client u receives the model of the most similar client and applies it to its data to predict landslide occurrence.

V. EVALUATION

A. Data Collection and Configuration

We have collected data related to landslide prediction for ten prefectures in Japan. Landslides depend on various features, and we have curated datasets with well-established and commonly used variables. **Landslide Events:** We collected records of landslide events in Japan from 2021 to 2022, totaling 253 events. Each landslide event is documented with its corresponding time and location, represented as six decimal points for latitude and longitude coordinates. The time granularity in our dataset is at the day level, and *SlideSafe* predicts landslide occurrences at this level, which is sufficient for pre-evacuation planning. **Rainfall:** As demonstrated in [4], time series data of rainfall contribute to accurate landslide prediction because water accumulation plays a crucial role in landslides. Our dataset contains 2 years of data from January 1st, 2021 to December 31, 2022. **Soil Property:** Soil water content is a significant factor contributing to landslides because underground water flow depends on soil properties. The properties related to the ease of sliding or water drainage have a significant impact on landslide events. In our dataset, the soil is categorized into 10 classes, including artificial paved areas. **Vegetation:** The

presence of vegetation in the surrounding area is a significant factor contributing to landslides, as plant and tree roots help stabilize underground water and soil. Our data sets include 11 types of vegetation. **Elevation and Slope Angle:** Elevation and slope angle are crucial factors in landslide prediction because they are closely linked to water accumulation and the gravitational force acting on the land. Additionally, steeper slopes are inherently more susceptible to landslides than gentler ones. Elevation and slope are represented as real numbers with eight categorized directions in our datasets.

We adhere to the Japan MESH3 Boundaries ECM (meshing system in Japan) and collect the data for each mesh as a unit, where the mesh length is one kilometer. To train and evaluate our system, we have prepared negative cases, representing days when no landslides occurred. For each client, we collected data for the days when current warning information was issued to ensure the quality of these negative cases. Additionally, we randomly selected an equal number of data points for each client to create negative cases. Furthermore, we show the experimental configuration settings in Table I.

B. Evaluation Metrics

The primary objective of *SlideSafe* is to achieve accurate landslide prediction. In disaster scenarios, achieving a recall rate of 100% is crucial because this information serves as the vital trigger for evacuations. However, it is also important to maintain a low false positive rate, as an excessive number of false alarms can erode trust in the information provided. To evaluate the practical utility of disaster information, we employ a performance metric known as *precision under 100% recall*. This metric allows us to assess the model's precision while ensuring that recall remains at 100%. Given that *SlideSafe* aims to predict landslides in distributed settings, all reported performance metrics are averaged across clients for each distribution, providing a comprehensive assessment of the system's effectiveness.

C. Decentralized Prediction Model Evaluation

In this section, to measure the performance of *SlideSafe*, we compare the proposed system with the most relevant state-of-the-art techniques: FedAvg [32], FedProx [22], and FedProx with fine-tuning (FedProx-FT) [25]. FedAvg [32] is the most classic federated learning method, which is the baseline of the training results. FedProx [22] is a widely used method to make the training process efficient in non-IID datasets. FedProx uses a regulation parameter μ to prevent the local

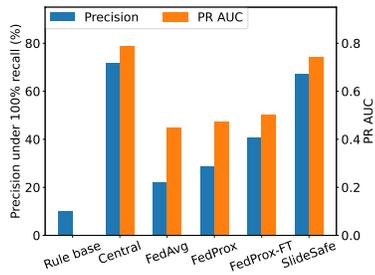


Fig. 6. Precision under 100% recall and precision and recall AUC (PR AUC)

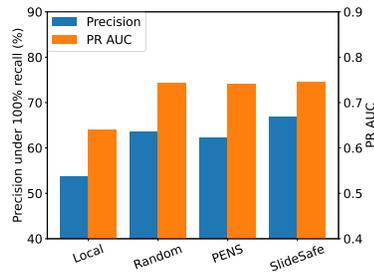


Fig. 7. Impact of client selection algorithm

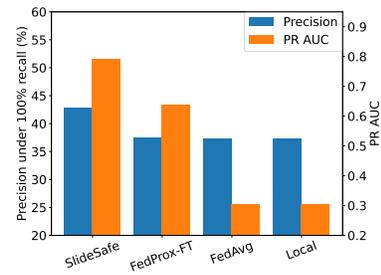


Fig. 8. Prediction performance in unseen region

model from deviating greatly from the global model. We fix the regulation parameter μ to 0.3. FedProx-FT [25] aims to build personalized models in non-IID settings by fine-tuning the global model trained in FedProx using local data. Furthermore, we set current Japanese warning information (Rule base) operated in the real world as a baseline.

1) *Impact of Similarity Threshold*: In *SlideSafe*, for each client, similar clients are selected to share the trained models, and this selection is controlled by m , the similarity threshold parameter. A smaller value of m results in grouping dissimilar clients and an increase in the impact of non-IID data distribution. Meanwhile, with a large value of m , the precision may also decrease because clients cannot collect models from sufficient clients to obtain comprehensive knowledge about landslides in decentralized collaborative learning. Therefore, we study the effect of m and consider the appropriate value through the experiment. Figure.3 shows the relationship between the threshold and the precision under 100% recall. We can observe that *SlideSafe* performs best with $m = 0.5$.

2) *Robustness*: *SlideSafe* aims to predict landslides in the real world. Such systems must be robust to tough situations because the settings and data are not necessarily ideal. We evaluate *SlideSafe*'s robustness from the number of clients. Generally, federated learning performs better as the number of clients increases. However, traditional federated learning methods struggle with non-IID data. Figure.5 presents the precision rates with different numbers of clients. Notably, we observe that *SlideSafe* and FedProx-FT [25] exhibit increasing performance trends as more clients are involved. It is worth noting that *SlideSafe* achieves higher performance with just three clients than FedProx-FT [25] does with ten clients. This highlights the efficacy of *SlideSafe*, stemming from its mechanism for grouping similar clients in federated learning.

The class imbalance ratio of our target dataset differs from clients (regions). Therefore, we observe the impact of class imbalance ratios on the precision. Figure 4 shows the results. Although all the methods are affected by the increase of the ratio, the decreasing trend of *SlideSafe* is much slower than the other two comparisons and the precision rates are much higher than those of the others. This indicates that *SlideSafe* is robust to class imbalance.

3) *Performance Comparison*: Figure 6 displays the precision under 100% recall and PR AUC of each method. In our experiments, we have ten local clients, and the values of

other relevant parameters can be found in Table I. *SlideSafe* takes 1.5 seconds to train the model for each epoch and 300ms to inference on average. Our machine environment is the following: GPU is NVIDIA GeForce RTX 4090, CPU is AMD Ryzen 9 7950X 16-Core Processor, and RAM is 64GB. Since *SlideSafe* works on local municipalities and needs only 300ms in runtime, local municipalities can make predictions in real time. It is worth noting that the centralized model in the graph utilizes data from all clients, making its performance a reference for an ideal and best-case scenario. In our observations, *SlideSafe* achieves the highest performance among the other state-of-the-art methods within decentralized settings. This outcome highlights that *SlideSafe* excels by personalizing models for target regions while acquiring comprehensive knowledge through decentralized collaborative learning with similar regions.

SlideSafe utilizes Personalized Adaptive Neighbor Matching (PANMGrad) [28] to identify clients with similar data distributions. To assess the impact of this approach, we compare *SlideSafe* with two others: random gossip (Random) and Performance-based node selection (PENS) [6]. In the random method, clients communicate with each other randomly, potentially leading to a mix of dissimilar data distributions. PENS is a state-of-the-art method for finding clients with similar data distributions. It selects similar clients by leveraging the loss values of client j 's model validated on client i 's local datasets. While PENS is a promising metric, it requires $O(N^2)$ communications to identify similar clients, where N is the number of clients in the network. Figure 7 presents precision under 100% recall and PR AUC for each method. Notably, *SlideSafe* outperforms other methods in terms of precision. This superior performance is attributed to *SlideSafe*'s successful formation of client groups with similar data distributions, enabling efficient knowledge sharing. It should also be highlighted that *SlideSafe* requires fewer communications, specifically $O(N)$ communications.

Finally, to assess the performance of *SlideSafe* in regions that have never experienced landslides, we examine its performance in unseen regions by applying a trained model to these unfamiliar areas. We select the model to use in these unseen regions using the PANMGrad approach. In *SlideSafe*, clients can identify similar clients without relying on labels, allowing us to apply the model in the region most similar to the target region. Figure 8 shows the model's performance in un-

familiar regions, which demonstrates *SlideSafe* predicts more accurately in unfamiliar regions than the existing methods.

VI. CONCLUSION

This paper presents a novel landslide prediction system combining spatio-temporal contrastive learning and selective collaborative learning. It begins by training a contrastive learning model to extract meaningful representations of land characteristics in each region. Subsequently, these trained models are merged among only regions with similar characteristics, leveraging collaborative learning. The federated models are then fine-tuned and customized for the landslide event prediction using the data specific to each region. We experimented with real-world datasets from ten Japanese prefectures. The results of these experiments illustrate that our approach achieves the highest precision when aiming for a 100% recall rate in landslide prediction, surpassing the performance of state-of-the-art methods. These findings underscore that our system performs well in alerting to signs of landslides while maintaining a low false-positive rate. In the future, we plan to boost the prediction reliability with explainable AI techniques (e.g., SHAP and LIME) and apply the framework to other types of disasters, such as floods and earthquakes.

ACKNOWLEDGMENT

This work was partially funded by JST, CREST Grant JPMJCR21M5, JSPS, KAKENHI Grant 22K12011 and 23H03384, and NVIDIA award. The authors appreciate JMA, MLIT, JMBSC, MOE, and GSI of Japan, which provide datasets to the authors. The JMA data used in this study was provided by way of the "Meteorological Research Consortium", a framework for research cooperation of JMA and MSJ.

REFERENCES

- [1] H. Rizk, Y. Nishimur, H. Yamaguchi, and T. Higashino, "Drone-based water level detection in flood disasters," *International journal of environmental research and public health*, vol. 19, no. 1, p. 237, 2021.
- [2] T. S. Teja, A. Dikshit, and N. Satyam, "Determination of rainfall thresholds for landslide prediction using an algorithm-based approach: Case study in the darjeeling himalayas, india," *Geosciences*, vol. 9, no. 7, 2019.
- [3] E. Collini, L. I. Palesi, P. Nesi, G. Pantaleo, N. Nocentini, and A. Rosi, "Predicting and understanding landslide events with explainable ai," *IEEE Access*, vol. 10, pp. 31 175–31 189, 2022.
- [4] P. Xie, A. Zhou, and B. Chai, "The application of long short-term memory (lstm) method on displacement prediction of multifactor-induced landslides," *IEEE Access*, vol. 7, pp. 54 305–54 311, 2019.
- [5] T. A. Steelman, S. M. McCaffrey, A.-L. K. Velez, and J. A. Briefel, "What information do people use, trust, and find useful during a disaster? evidence from five large wildfires," *Natural Hazards*, vol. 76, 2015.
- [6] N. Onoszko, G. Karlsson, O. Mogren, and E. L. Zec, "Decentralized federated learning of deep neural networks on non-iid data," 2021.
- [7] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 10 713–10 722.
- [8] Z. Liu, D. Guo, S. Lacasse, J.-h. Li, B. Yang, and J. C. Choi, "Algorithms for intelligent prediction of landslide displacements," 2020.
- [9] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data*, vol. 6, no. 1, pp. 1–54, 2019.
- [10] E. Tasci, Y. Zhuge, K. Camphausen, and A. V. Krauze, "Bias and class imbalance in oncologic data—towards inclusive and transferrable ai in large scale oncology data sets," *Cancers*, vol. 14, no. 12, p. 2897, 2022.
- [11] A. A. Alsau, Y. A. Alghofaili, M. Alghadeer, and F. H. Alharbi, "Resampling techniques for materials informatics: limitations in crystal point groups classification," *Journal of Chemical Information and Modeling*, vol. 62, no. 15, pp. 3514–3523, 2022.
- [12] A. Singh, R. K. Ranjan, and A. Tiwari, "Credit card fraud detection under extreme imbalanced data: a comparative study of data-level algorithms," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 34, no. 4, pp. 571–598, 2022.
- [13] B. Krawczyk, C. Bellinger, R. Corizzo, and N. Japkowicz, "Under-sampling with support vectors for multi-class imbalanced data classification," in *2021 International Joint Conference on Neural Networks (IJCNN)*, 2021, pp. 1–7.
- [14] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.
- [15] D. Dablain, B. Krawczyk, and N. V. Chawla, "Deepsmote: Fusing deep learning and smote for imbalanced data," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- [16] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, 2018, pp. 117–122.
- [17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [18] F. Feng, K.-C. Li, J. Shen, Q. Zhou, and X. Yang, "Using cost-sensitive learning and feature selection algorithms to improve the performance of imbalanced classification," *IEEE Access*, vol. 8, 2020.
- [19] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy, "Training deep neural networks on imbalanced data sets," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 4368–4374.
- [20] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.
- [21] I. Hegedűs, G. Danner, and M. Jelasity, "Decentralized learning works: An empirical comparison of gossip learning and federated learning," *Journal of Parallel and Distributed Computing*, vol. 148, 2021.
- [22] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proceedings of Machine Learning and Systems*, I. Dhillon, D. Papailiopoulos, and V. Sze, Eds., vol. 2, 2020, pp. 429–450.
- [23] D. Rothchild, A. Panda, E. Ullah, N. Ivkin, I. Stoica, V. Braverman, J. Gonzalez, and R. Arora, "Fetchsgd: Communication-efficient federated learning with sketching," in *International Conference on Machine Learning*. PMLR, 2020, pp. 8253–8265.
- [24] N. Dong and I. Voiculescu, "Federated contrastive learning for decentralized unlabeled medical images," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2021, pp. 378–387.
- [25] K. Wang, R. Mathews, C. Kiddon, H. Eichner, F. Beaufays, and D. Ramage, "Federated evaluation of on-device personalization," 2019.
- [26] Y. Huang, L. Chu, Z. Zhou, L. Wang, J. Liu, J. Pei, and Y. Zhang, "Personalized cross-silo federated learning on non-iid data," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 9, 2021.
- [27] R. Dai, L. Shen, F. He, X. Tian, and D. Tao, "Dispfl: Towards communication-efficient personalized federated learning via decentralized sparse training," *arXiv preprint arXiv:2206.00187*, 2022.
- [28] Z. Li, J. Lu, S. Luo, D. Zhu, Y. Shao, Y. Li, Z. Zhang, Y. Wang, and C. Wu, "Towards effective clustered federated learning: A peer-to-peer framework with adaptive neighbor matching," *IEEE Transactions on Big Data*, pp. 1–16, 2022.
- [29] A. Amini, M. Dolatshahi, and R. Kerachian, "Adaptive precipitation nowcasting using deep learning and ensemble modeling," *Journal of Hydrology*, vol. 612, p. 128197, 2022.
- [30] L. Wang, S. Xu, X. Wang, and Q. Zhu, "Addressing class imbalance in federated learning," 2020.
- [31] Z. Liu, S. Luo, W. Li, J. Lu, Y. Wu, S. Sun, C. Li, and L. Yang, "Convtransformer: A convolutional transformer network for video frame synthesis," 2021.
- [32] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. PMLR, 20–22 Apr 2017.