Landslide Detection in Real-Time Social Media Image Streams

Ferda Ofli[1*], Muhammad Imran[1], Umair Qazi[1], Julien Roch[2], Catherine Pennington[3], Vanessa Banks[3] and Remy Bossu[2,4]

[1*]Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha, 34110, Qatar.
[2]European-Mediterranean Seismological Centre, Arpajon, 91297, France.
[3]British Geological Survey, Keyworth, Nottinghamshire, NG12 5GG, United Kingdom.
[4]CEA, DAM, DIF, Arpajon, 91297, France.

*Corresponding author(s). E-mail(s): fofli@hbku.edu.qa;
Contributing authors: mimran@hbku.edu.qa; uqazi@hbku.edu.qa; julien.roch@emsc-csem.org; cpoulton@bgs.ac.uk; vbanks@bgs.ac.uk; bossu@emsc-csem.org;

Abstract

Lack of global data inventories obstructs scientific modeling of and response to landslide hazards which are oftentimes deadly and costly. To remedy this limitation, new approaches suggest solutions based on citizen science that requires active participation. In contrast, as a non-traditional data source, social media has been increasingly used in many disaster response and management studies in recent years. Inspired by this trend, we propose to capitalize on social media data to mine landslide-related information automatically with the help of artificial intelligence techniques. Specifically, we develop a state-of-the-art computer vision model to detect landslides in social media image streams in real time. To that end, we first create a large landslide image dataset labeled by experts with a data-centric perspective, and then, conduct extensive model training experiments. The experimental results indicate that the proposed model can be deployed in an online fashion to support global landslide susceptibility maps and emergency response.

Keywords: Landslide detection, Social media, Image classification, Data-centric AI

1 Introduction

Landslides[1] occur all around the world and cause thousands of deaths and billions of dollars in infrastructural damage worldwide every year [1]. However, landslide events are often under-reported and insufficiently documented due to their complex natural phenomena governed by various intrinsic and external conditioning and triggering factors such as earthquakes and tropical storms, which are usually more conspicuous, and hence, more widely reported [2]. Due to this oversight and lack of global data inventories to study landslides, Froude and Petley assert that any attempt to quantify global landslide hazards and the associated impacts is destined to be an underestimation [3].

Existing landslide detection and mapping solutions typically rely on data from ground sensors or satellites. While sensor-based approaches can achieve high accuracy at sub-catchment levels by monitoring land characteristics such as rainfall, altitude, soil type, and slope [4, 5], their global-scale deployment is impractical. Satellite-based approaches can provide more scalable solutions by analyzing Synthetic Aperture Radar (SAR) or optical imagery [6, 7]. However, their deployment can still prove costly and time-consuming. Furthermore, satellite data is susceptible to noise such as clouds.

Using Volunteered Geographical Information (VGI) as an alternative approach, NASA launched a website[2] in 2018 to allow citizens to report about the regional landslides they see in-person or online [8]. Subsequent studies developed other means such as mobile apps to collect citizen-provided data [9, 10]. However, these studies assume active participation of volunteers to collect landslide data and still require time consuming work by specialists directly engaging with the volunteers and interpreting the received data [11].

To alleviate the need for opt-in participation and manual processing, we develop a state-of-the-art AI model that can automatically detect landslides from social media images in real time. To achieve this goal, we first create a large image dataset comprising 11,737 images from various data sources annotated by domain experts following a data-centric AI approach described by Whang et al. [12]. We then exploit this dataset in a comprehensive experimentation searching for the optimal landslide model configuration (as in [13, 14]). This exploration reveals interesting insights about the model training process. The optimal landslide model achieves an accuracy of 90.6% on the validation set, 87.0% on the held-out test set, and a striking 97.7% when applied on the real-time Twitter image stream *in the wild*. Based on this model, we envision

[1] We refer to all downward and outward movement of loosen slope materials such as landslip, debris flows, mudslides, rockfalls, earthflows, and other mass movements as landslides in this study.
[2] https://gpm.nasa.gov/landslides/index.html

a system that can harvest global landslide data and facilitate further research for building global landslide susceptibility maps as suggested in [15, 16].

We make the following contributions:

- We collected the largest dataset of ground-level landslide images to date.
- We followed a data-centric AI approach to iteratively improve the quality of the dataset.
- We conducted the most comprehensive experiments to date for training deep learning models for landslide recognition.
- We built a prototype system and deployed our landslide detection model in the real world to assess its performance *in the wild*.
- The prototype system offers global scalability by leveraging social media data as a form of passive (i.e., opportunistic) crowdsourcing.

The rest of the paper is organized as follows. Section 2 reviews the relevant literature, Section 3 introduces the dataset, Section 4 describes the model training experiments, Section 5 summarizes the experimental results and findings, Section 7 provides a discussion on existing limitations and future work, and finally, Section 8 concludes the paper.

2 Related Work

The literature on landslide detection and mapping approaches mainly uses four types of data sources: *(i) physical sensors*, *(ii) remote sensing*, *(iii) volunteers*, and *(iv) social networks*. Sensor-based approaches rely on land characteristics such as rainfall, altitude, soil type, and slope to detect landslides and develop models to predict future events [4, 5]. While these approaches can be highly accurate at sub-catchment levels, their large-scale deployment is extremely costly.

Earth observation data obtained using high-resolution satellite imagery has been widely used for landslide detection, mapping, and monitoring [6, 7]. Remote sensing techniques either use Synthetic Aperture Radar (SAR) or optical imagery to identify landslides following various approaches from image classification [17, 18] and segmentation [19, 20] to object detection [21, 22]. While remote sensing through satellites can be useful to monitor landslides globally, their deployment can prove costly and time-consuming. Moreover, satellite data is susceptible to noise such as clouds.

A few studies demonstrate the use of Volunteered Geographical Information (VGI) as an alternative method to detect landslides [9, 23–25]. These studies assume active participation of volunteers to collect landslide data where the volunteers opt in to use a mobile app to provide information such as photos, time of occurrence, damage description and other observations about a landslide event. In order to validate landslide photos collected by the volunteers, Can et al. present an image classification model based on Convolutional Neural Networks (CNN) trained on a relatively small in-house dataset [24]. On the contrary, our work aims to capitalize on massive social media data without any active participation requirement and with better scalability. In addition,

we construct a much larger dataset to train deep learning models and perform more extensive experimental evaluations.

Social media data has been used in many humanitarian contexts ranging from general social analytics [26] and geospatial sentiment analysis [27] to incident detection [28] and rapid damage assessment [29], including multimodal approaches [30]. However, its use for landslide detection has not been explored extensively. To the best of our knowledge, no prior work has explored the use of social media imagery to detect landslides. The most relevant studies by Musaev et al. combine social media text data and physical sensors to detect landslides [31, 32]. Specifically, they use textual messages collected through a set of landslide-related keywords on Twitter, Instagram, and YouTube in combination with sensor data about seismic activity and rainfall to train a machine learning classifier that can identify landslide incidents. In this study, we focus on analyzing social media images which can provide more detailed information about the impact of the landslide event. To that end, our work is orthogonal to prior art.

Finally, this paper is different from and complementary to our previous papers [15, 16] in the following ways. In [15], we present a narrative from a practitioner perspective that predominantly highlights existing limitations and challenges in landslide research and proposes a high-level methodology including data collection, processing, and annotation for an AI-based solution without going into technical details of the machine learning aspects of the problem. In [16], we focus on the system engineering aspects where we present building blocks of an online system that can ingest social media data, eliminate duplicate and irrelevant content as well as identify and geolocate landslide reports. We also provide proper latency and throughput benchmark results for each system component. The landslide detection model is covered very briefly in this context. In this paper, on the other hand, we elaborate on all the technical details about the machine learning model development aspects of the problem through an extensive experimentation in search for the optimal model selection and training configuration. To ensure the paper is self-contained, we recapitulate the most relevant parts of our prior works here very briefly.

3 Dataset

To train models that can detect landslides in images, we curated a large image dataset from multiple sources with diverse characteristics. We collected some images from the Web using Google Image search with keywords such as *landslide, landslip, earth slip, mudslide, rockslide, rock fall* and some images from Twitter using similar landslide-related hashtags. We obtained additional images from landslide specialists captured during field trips. The images obtained from social media or the Web are usually noisy and can include duplicates. Similarly, the images captured during field trips are not always useful for model training. Therefore, the collected data is manually labeled by three landslide experts, who are also co-authors of this study, following a data-centric

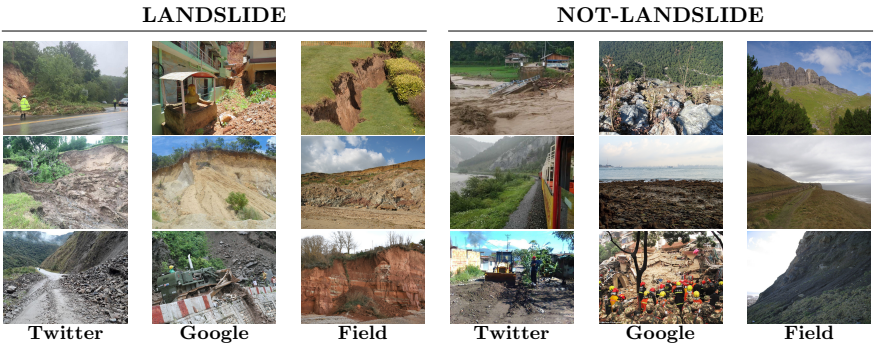


Fig. 1: Example images from the dataset

AI [12] approach that suggests focusing on the data pipeline which typically involves (i) curating a dataset for labeling based on model performance after every iterative cycle to address the model's specific weaknesses and (ii) significantly increasing performance with a relatively small amount of training data, as elaborated in [15]. Since the AI task at hand is "given an image, recognize landslides" (i.e., no other external information or expert knowledge is available to the AI model), the experts were instructed to keep this *computer-vision perspective* in mind and label only the most evident cases as "landslide" images (i.e., the images where the landslide is the main theme exhibiting substantial visual cues for the model to learn from). On the other hand, since our ultimate goal is to develop a system that will continuously monitor the noisy social media streams to detect landslide events in real time, we retained *negative* (i.e., not-landslide) images that illustrate completely irrelevant cases (e.g., cartoons, advertisements, selfies) as well as difficult scenarios such as post-disaster images from earthquakes and floods in addition to other natural scenes without landslides in the final dataset. The complete dataset creation process includes several rounds of model training, error analysis, expert discussions, and label updates. The final dataset contains 11,737 images. Some example images are shown in Fig. 1. The distribution of images across data sources is summarized in Table 1 and the data splits are presented in Table 2. As suggested by Table 2, only about 23% of the images are categorized as "landslide." Our dataset is currently the largest dataset for landslide recognition from ground-level images. To assess the quality of the final labels, we measured the inter-annotator agreement using two statistical measures: Fleiss' Kappa [33] and percentage agreement (observer agreement). Despite the inherent difficulty of the task, the experts achieved an overall Fleiss' Kappa of 0.58, which indicates an almost substantial inter-annotator agreement. They also achieved a percentage agreement of 76%, which is only slightly below the 80% mark set as a rule-of-thumb by Bayerl and Paul [34].

Table 1: Distribution of images across data sources

	Training	Validation	Test	Total
Google	4,398	628	1,258	6,284
Twitter	807	115	231	1,153
Field	3,010	430	860	4,300
Total	8,215	1,173	2,349	11,737

Table 2: Data splits (70:10:20)

	Training	Validation	Test	Total
Landslide	1,883	271	536	2,690
Not-landslide	6,332	902	1,813	9,047
Total	8,215	1,173	2,349	11,737

4 Landslide Model

Many computer vision tasks have greatly benefited from the recent advances in deep learning. The features learned in deep convolutional neural networks (CNNs) are proven to be transferable and quite effective when used in other visual recognition tasks [35–37], particularly when training samples are limited. Considering we also have limited training examples for data-hungry deep CNNs, we follow a transfer learning approach to adapt the features and parameters of the network from the broad domain (i.e., large-scale image classification) to the specific one (i.e., landslide classification). However, it is often overlooked how complex the transfer learning setup can become with all different possible configurations and hyperparameters to tune for optimal performance. To this end, [13, 14] present exemplary studies on empirical analysis of the impact of different training strategies on the performance of ResNet architecture where they explore training recipes with different loss functions, data augmentation, regularization, and optimization techniques, among others. Inspired by these studies, we conduct extensive experiments where we train several different deep CNN architectures using different optimizers, learning rates, weight decays, and class balancing strategies.

CNN Architecture. The CNN architecture (arch) plays a significant role on the performance of the resulting model depending on the available data size and problem characteristics. Therefore, we explored a representative sample of well-known CNN architectures including VGG16 [38], ResNet18, ResNet50, ResNet101 [39], DenseNet [40], InceptionNet [41], and EfficientNet [42], among others.

Optimizer. An optimizer (opt) is an algorithm or method that changes the attributes of a neural network (e.g., weights and learning rate) in order to

reduce the optimization loss and to increase the desired performance metric (e.g., accuracy). In this study, we experimented with the most popular optimizers, i.e., Stochastic Gradient Descent (SGD) and Adam [43].

Learning rate. Learning rate (lr) controls how quickly the model is adapted to the problem. Using a too large learning rate can cause the model to converge too quickly to a suboptimal solution whereas a too small learning rate can cause the process to get stuck. Since learning rate is one of the most important hyperparameters and setting it correctly is critical for real-world applications, we performed a grid search over a large range of values (i.e., $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$).

Weight decay. Weight decay (wd) controls the regularization of the model weights, which in turn, helps to avoid overfitting of a deep neural network on the training data and improve the performance of the model on the unseen data (i.e., better generalization ability). In light of this, we experimented with a large range of weight decay values (i.e., $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$).

Class balancing. An imbalanced dataset can bias the prediction model towards the dominant class (i.e., not-landslide) and lead to poor performance on the minority class (i.e., landslide), which is not ideal for our application. The approaches to tackle this problem range from generating synthetic data to using specialized algorithms and loss functions. Here, we explored one of the basic approaches, i.e., data resampling, where we oversampled images from the landslide class (i.e., sampling with replacement) to create a balanced training set.

Other training details. We ran all our experiments on Nvidia Tesla P100 GPUs with 16GB memory using PyTorch library.[3] We adjusted the batch size according to each CNN architecture in order to maximize GPU memory utilization. We used a fixed step size of 50 epochs in the learning rate scheduler of the SGD optimizer and a fixed patience of 50 epochs in the *'ReduceLROnPlateau'* scheduler of the Adam optimizer, both with a factor of 0.1. All of the models were initialized using the weights pretrained on ImageNet [44] and trained for a total of 200 epochs. Consequently, we trained a total of 560 CNN models in our quest for the best model configuration.

5 Results

Due to limited space, Table 3 presents results only for the top performing 10 model configurations on the validation set ranked based on Matthew Correlation Coefficient (MCC), which is regarded as a balanced measure for imbalanced classification problems [45] and defined by Equation 1.

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \tag{1}$$

[3] https://pytorch.org/

Table 3: Top-10 configurations based on MCC on the validation set

Opt	Arch	CB	LR	WD	Acc	Prec	Rec	F1	MCC
Adam	ResNet50	✗	10^{-4}	10^{-3}	0.906	0.821	0.760	0.789	0.730
SGD	ResNet101	✗	10^{-3}	10^{-5}	0.905	0.835	0.731	0.780	0.722
SGD	ResNet101	✓	10^{-2}	10^{-5}	0.904	0.821	0.745	0.781	0.721
SGD	ResNet50	✗	10^{-3}	10^{-3}	0.905	0.838	0.727	0.779	0.721
SGD	DenseNet	✗	10^{-2}	10^{-4}	0.902	0.800	0.768	0.783	0.720
Adam	ResNet50	✗	10^{-4}	10^{-2}	0.903	0.834	0.723	0.775	0.716
SGD	ResNet50	✓	10^{-2}	10^{-5}	0.903	0.834	0.723	0.775	0.716
SGD	EfficientNet	✗	10^{-2}	10^{-3}	0.897	0.768	0.793	0.780	0.713
Adam	ResNet101	✗	10^{-4}	10^{-3}	0.902	0.845	0.705	0.769	0.712
Adam	ResNet101	✓	10^{-4}	10^{-5}	0.899	0.802	0.745	0.772	0.708

where TP is the number of true positives, TN the number of true negatives, FP the number of false positives, and FN the number of false negatives. Besides MCC, we also compute common performance metrics such as Accuracy, Precision, Recall, and F1-score as defined by Equations 2-5, respectively.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{2}$$

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{4}$$

$$\text{F1} = \frac{2TP}{2TP + FP + FN}. \tag{5}$$

The top-performing model configuration (i.e., arch: ResNet50, opt: Adam, lr: 10^{-4}, wd: 10^{-3}, no class balancing) achieves MCC=0.730, F1=0.789, and Accuracy=0.906, all deemed plausible by the specialists. Nevertheless, we investigate the full table of results and identify the following insights:

- When everything but the optimizer is kept fixed, the models trained with the Adam optimizer outperforms the models trained with the SGD optimizer (179 vs. 100). This confirms the general sentiment that the adaptive and stable nature of the Adam optimizer necessitates less effort to achieve convergence and attain superior training outcomes than the SGD optimizer.
- Despite the fact that top-performing model is trained without a class balancing strategy, the overall trend indicates that, while everything else is the same, the models trained with class balancing yield better performance than those trained without class balancing (173 vs. 103). This is inline with the general understanding that class balancing can prevent the models from becoming biased towards the majority class, and hence, generate higher accuracy models.

Table 4: Performance comparison of CNN architectures

Architecture	mean(MCC)	std(MCC)	Avg. Rank
ResNet50	0.5384	0.2059	2.7625
ResNet101	0.5350	0.1975	2.9875
VGG16	0.5267	0.2026	3.2125
DenseNet	0.5219	0.1993	3.6125
EfficientNet	0.4951	0.2267	4.0625
ResNet18	0.4956	0.2065	4.7000
InceptionNet	0.3516	0.1758	6.6625

- ResNet50 architecture tops the rankings among all CNN architectures by achieving the best average ranking as well as the highest mean MCC according to Table 4. Between the ResNet architectures, given that the training dataset is relatively small, ResNet18 offers inadequate capacity for the problem at hand whereas ResNet101 offers potentially more-than-enough capacity which increases the risk of overfitting and hurts the performance. However, the overall differences between architectures do not seem significant except for InceptionNet which yields a significantly poorer performance than others. This is potentially because the InceptionNet architecture generally requires more data to overcome possible overfitting and more computational resources.
- The impact of the learning rate on model performance shows opposite trends for different optimizers. As per Table 5, smaller learning rates (e.g., $\{10^{-6}, 10^{-5}, 10^{-4}\}$) seem to work better with the Adam optimizer whereas larger learning rates (e.g., $\{10^{-2}, 10^{-3}\}$) seem to work better with the SGD optimizer. This is because when the SGD optimizer is initialized with a very small learning rate, the training progress becomes very slow and tends to stagnate at a sub-optimal local minimum due to the scheduled learning rate updates at regular intervals. In contrast, the Adam optimizer typically operates better with a smaller learning rate since it ensures a more stable adaptation during training.
- As expected, the value of the weight decay also impacts the overall performance significantly (in particular, for the Adam optimizer). A large weight decay (e.g., 10^{-2}) hurts the overall performance which tends to improve as the weight decay takes on smaller values (see Table 6). This implies that larger weight decay values cause excessive regularization of the weights, which in turn, reduces the model's ability to learn properly.

To illustrate the effectiveness of the transfer learning approach, we created t-SNE [46] visualizations of the feature embeddings before and after the training of the best-performing model. As shown in Fig. 2, the original ResNet50 model pretrained on ImageNet can distinguish landslide from not-landslide images neither in the training (Fig. 2a) nor in the validation set (Fig. 2b). However, after finetuning the model on the target landslide dataset, the resulting feature embeddings show almost perfect separation of the classes in the

Table 5: Effect of the learning rate on overall performance

Adam		Learning Rate	SGD	
(mean)	(std)		(mean)	(std)
0.5812	0.0660	10^{-6}	0.0947	0.1239
0.6077	0.0708	10^{-5}	0.3335	0.1825
0.6495	0.0725	10^{-4}	0.5597	0.0904
0.5438	0.1223	10^{-3}	0.6287	0.0710
0.3178	0.2026	10^{-2}	0.6325	0.0822

Table 6: Effect of the weight decay on overall performance

Adam		Weight Decay	SGD	
(mean)	(std)		(mean)	(std)
0.5772	0.1270	10^{-5}	0.4586	0.2369
0.5685	0.1284	10^{-4}	0.4594	0.2368
0.5462	0.1409	10^{-3}	0.4555	0.2441
0.4681	0.2263	10^{-2}	0.4258	0.2415

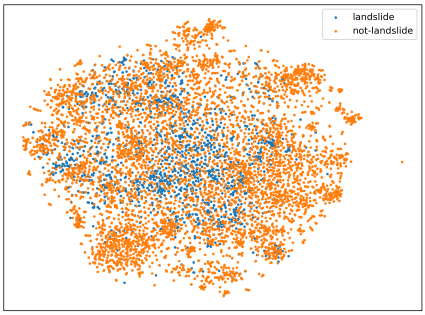
Table 7: Performance comparison of the best model on the validation and test sets

Set	Accuracy	Precision	Recall	F1-score	MCC
Validation	0.906	0.821	0.760	0.789	0.730
Test	0.870	0.737	0.668	0.701	0.619

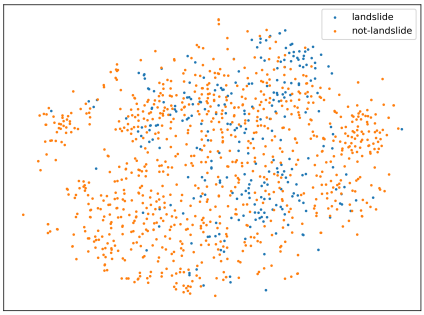
training set (Fig. 2c) and a reasonably well separation in the validation set (Fig. 2d).

When applied on the held-out test set, the best-performing model achieves MCC=0.619, F1=0.701, and Accuracy=0.870 as opposed to MCC=0.730, F1=0.789, and Accuracy=0.906 achieved on the validation set (Table 7). Although the difference in accuracy is relatively small, the difference in MCC and F1 are considerably large due to significant drops in precision and recall of the model on the test set. This phenomenon can be explained by the more-than-twice increase in the false positive (128 vs. 45) and false negative (178 vs. 65) predictions of the model on the test set, potentially as a result of model overfitting to the validation set (Table 8).

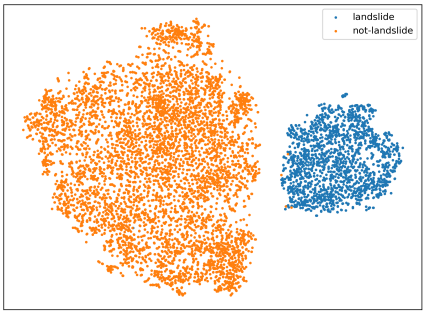
To have a better understanding of the inner workings of the model, we investigated class activation maps [47], which highlight the discriminative image regions that the CNN model pays attention to decide whether an image belongs to landslide or not-landslide class. Fig. 3 demonstrates example visualizations for all four cases, i.e., true positives, true negatives, false positives, and false negatives. The visualizations for the true positive predictions indicate that the model successfully localizes the landslide regions (e.g., rockfalls,



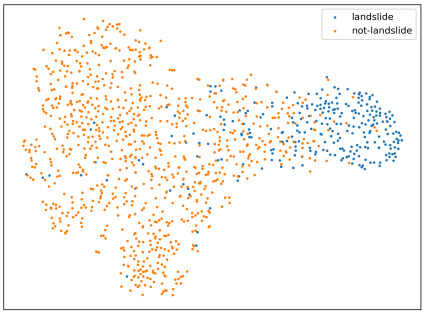
(a) Training set before finetuning



(b) Validation set before finetuning



(c) Training set after finetuning



(d) Validation set after finetuning

Fig. 2: Feature embeddings before/after model finetuning

Table 8: Confusion matrices for the validation and test sets

		Prediction	
	Ground Truth	Landslide	Not-landslide
Validation (10%)	Landslide	206	65
	Not-landslide	45	857
Test (20%)	Landslide	358	178
	Not-landslide	128	1,685

earthslip, etc.) in the images. Similarly for the true negative predictions, the model focuses on areas that do not show any landslide cues, successfully avoiding tricky conditions such as muddy roads, wet surfaces, and natural rocky areas on a beach. However, in both false positive and false negative predictions, we observe that the errors occur mainly because the model fails to localize its attention on a particular region in the image, or is tricked by the image regions that are reminiscent of landslide scenes. This analysis suggests that there is room for improvement where we can train more robust models by enriching

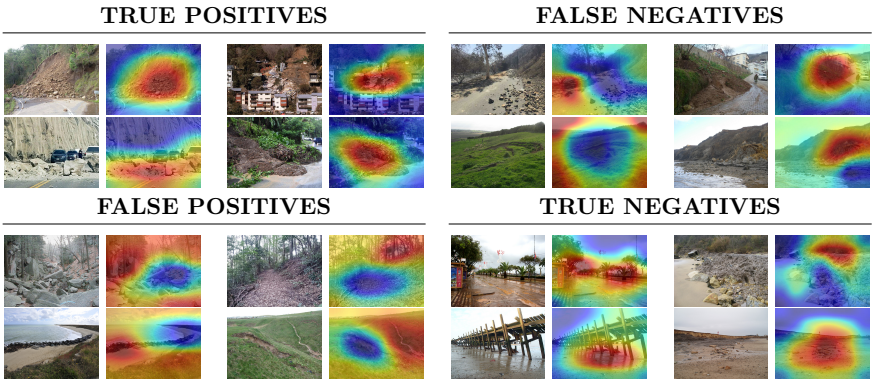


Fig. 3: Class activation maps of the model predictions on the test set.

the training set with additional hard negative and hard positive images. For instance, we can add more images of forest areas without any landslides to reduce false positives and more images of small-scale landslides to reduce false negatives.

6 Real-World Deployment

We have developed a proof-of-concept system as presented in [16]. In a nutshell, the system (i) collects live tweets from the Twitter Streaming API[4] that match landslide-related keywords in multiple languages, (ii) extracts image URLs from the tweets (if any) and downloads images, (iii) runs the downloaded images through filtering models to eliminate duplicate and irrelevant content, (iv) runs the remaining images through the landslide model to tag each image as landslide or not-landslide, and finally, (v) displays the results on a dashboard for specialists' examination. The system has collected almost 4.5 million images since its deployment in February 2020. However, only about 30,000 images have been labeled as landslide, which corresponds to less than 1% of the total volume. This indicates the difficulty of the task even though a carefully curated set of landslide-related keywords has been used to collect data from Twitter. To validate the performance of the landslide model *in the wild*, the specialists reviewed a random subset of the collected images (N=3,600) and assigned ground truth labels. We then re-computed performance scores for the real-world evaluation of the model (Table 9). Satisfactorily, the model achieves a comparable performance to our experiments, and more importantly, generalizes well to a challenging real-world scenario.

7 Discussion

[4] https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data

Table 9: Evaluation of the real-world performance

TP	FP	FN	TN	TOTAL
123	39	43	3,395	3,600
Accuracy	**Precision**	**Recall**	**F1-score**	**MCC**
0.977	0.759	0.741	0.750	0.738

Our experimental results and analytical findings suggest that CNN-based image classification models, when tuned optimally, can be useful for the challenging task of recognizing landslides from images. More importantly, instead of depending on citizen science projects (i.e., active crowdsourcing), we can scale up the solution much more efficiently by relying on passive crowdsourcing and leveraging the information shared in online social media platforms. This ability paves the way for an AI-based automated system that can monitor landslide events around the world, and eventually, reduce human effort and operational cost. Hence, we believe the contributions of the current study will advance the state of art in global landslide data and research. However, we also acknowledge that there are some limitations of the current study. Below we elaborate on the implications of our experimental findings, existing limitations, and our future work in more detail.

On the technical side, it is important to note that our comprehensive experimentation focused exclusively on a selection of CNN architectures. However, transformer-based models, e.g. Vision Transformer (ViT) [48], have recently become more popular and shown to outperform their CNN counterparts in various computer vision tasks. Therefore, it is expected that transformer-based image classification models can lead to better landslide detection performances. Besides, we did not explore thoroughly the effect of stronger data augmentation (e.g., RandAugment [49] and CutMix [50]) and regularization (e.g., label smoothing [41] and dropout [51]) in our current setup to keep the computational workload at a manageable level. Hence, it might be possible to improve the model performance further via stronger data augmentation and regularization techniques, as well. We suggest running an extended experimentation to evaluate state-of-the-art vision transformer models as future work. Another potential extension of our work can be around multimodal modeling of social media text and images together for landslide detection as suggested in [52].

On the application side, despite the fact that social media platforms provide quick access to situational information during time-critical events, we note that a large portion of this data contains irrelevant and redundant information. Therefore, tasking a *single* model (i.e., landslide model) to sift through all the noise in the social media data alone might not be a plausible system realization. Instead, it is advisable to support the landslide model with other image classification models for filtering out duplicate and irrelevant content, as implemented in [16]. Similarly, current study does not evaluate the authenticity and veracity of the landslide images collected from social media. We

believe this requires further investigation through other automatic or manual processes. It is important to reiterate that this work is not intended to be used in isolation during a disaster scenario. As well as the inherent noise within the data content itself, there are inaccuracies that could, in the worst case, hinder rescue operations if not combined with other data sources.

8 Conclusion

In this study, we developed a model that can automatically detect landslides in social media image streams. For this purpose, we first created a large image collection from multiple sources with different characteristics to ensure data diversity. Then, the collected images were assessed by three experts to attain high quality labels through an iterative process of data re-labeling and model retraining as per data-centric AI principles. The collected dataset is currently the largest dataset for landslide recognition from ground-level images. At the heart of this study lied an extensive search for the optimal landslide model configuration with various CNN architectures, network optimizers, learning rates, weight decays, and class balancing strategies. We provided several insights about the impact of each optimization dimension on the overall performance. These insights validated common practices and expectations shared by the community through controlled experiments in one place. The best-performing model achieved plausible performance not only under an experimental setup but also *in the wild* during a real-world deployment. This underlines the feasibility of our ultimate goal—building a system that leverages social media data as a form of passive (i.e., opportunistic) crowdsourcing to detect landslide reports in real time and at scale. We believe such a system can contribute to harvesting of global landslide data and facilitate further landslide research. More importantly, it can support global landslide susceptibility maps to provide situational awareness and improve emergency response and decision making.

Declarations

Funding This article was partially funded by the European Union's (EU) Horizon 2020 Research and Innovation Program under Grant Agreement RISE Number 821115. Opinions expressed in this article solely reflect the authors' views; the EU is not responsible for any use that may be made of information it contains. The British Geological Survey (UK Research and Innovation) granted supporting research funding through National Capability (Shallow Geohazards) and Innovation funding streams. Open access funding information will be available upon approval.

Competing interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability The dataset generated and analyzed during the current study is not publicly available due to privacy and other restrictions but are available from the corresponding author upon reasonable request.

References

[1] Kjekstad, O., Highland, L.: In: Sassa, K., Canuti, P. (eds.) Economic and Social Impacts of Landslides, pp. 573–587. Springer, Berlin, Heidelberg (2009). https://doi.org/10.1007/978-3-540-69970-5_30. https://doi.org/10.1007/978-3-540-69970-5_30

[2] Lee, E.M., Jones, D.K.C.: Landslide Risk Assessment. Thomas Telford Publishing, London (2004). https://doi.org/10.1680/lra.31715. https://www.icevirtuallibrary.com/doi/abs/10.1680/lra.31715

[3] Froude, M.J., Petley, D.N.: Global fatal landslide occurrence from 2004 to 2016. NHESS **18**(8), 2161–2181 (2018)

[4] Merghadi, A., Yunus, A.P., Dou, J., Whiteley, J., ThaiPham, B., Bui, D.T., Avtar, R., Abderrahmane, B.: Machine learning methods for landslide susceptibility studies: A comparative overview of algorithm performance. Earth-Science Reviews **207**, 103225 (2020)

[5] Ramesh, M.V., Kumar, S., Rangan, P.V.: Wireless sensor network for landslide detection. In: ICWN, pp. 89–95 (2009)

[6] Mondini, A.C., Guzzetti, F., Chang, K.-T., Monserrat, O., Martha, T.R., Manconi, A.: Landslide failures detection and mapping using synthetic aperture radar: Past, present and future. Earth-Science Reviews **216**, 103574 (2021). https://doi.org/10.1016/j.earscirev.2021.103574

[7] Mohan, A., Singh, A.K., Kumar, B., Dwivedi, R.: Review on remote sensing methods for landslide detection using machine and deep learning. Trans ETT **32**(7), 3998 (2021)

[8] Juang, C.S., Stanley, T.A., Kirschbaum, D.B.: Using citizen science to expand the global map of landslides: Introducing the cooperative open online landslide repository (coolr). PloS one **14**(7), 0218657 (2019)

[9] Kocaman, S., Gokceoglu, C.: A CitSci app for landslide data collection. Landslides **16**(3), 611–615 (2019)

[10] Cieslik, K., Shakya, P., Uprety, M., Dewulf, A., Russell, C., Clark, J., Dhital, M.R., Dhakal, A.: Building resilience to chronic landslide hazard through citizen science. Frontiers in Earth Science **7**, 278 (2019)

[11] Pennington, C., Freeborough, K., Dashwood, C., Dijkstra, T., Lawrie,

K.: The national landslide database of great britain: Acquisition, communication and the role of social media. Geomorphology **249**, 44–51 (2015)

[12] Whang, S.E., Roh, Y., Song, H., Lee, J.-G.: Data collection and quality challenges in deep learning: A data-centric AI perspective. The VLDB Journal, 1–23 (2023)

[13] Bello, I., Fedus, W., Du, X., Cubuk, E.D., Srinivas, A., Lin, T.-Y., Shlens, J., Zoph, B.: Revisiting ResNets: Improved training and scaling strategies. In: NeurIPS (2021)

[14] Wightman, R., Touvron, H., Jégou, H.: ResNet strikes back: An improved training procedure in timm. arXiv:2110.00476 (2021)

[15] Pennington, C.V.L., Bossu, R., Ofli, F., Imran, M., Qazi, U., Roch, J., Banks, V.J.: A near-real-time global landslide incident reporting tool demonstrator using social media and artificial intelligence. International Journal of Disaster Risk Reduction **77**, 103089 (2022). https://doi.org/10.1016/j.ijdrr.2022.103089

[16] Ofli, F., Qazi, U., Imran, M., Roch, J., Pennington, C., Banks, V., Bossu, R.: A real-time system for detecting landslide reports on social media using artificial intelligence. In: Di Noia, T., Ko, I.-Y., Schedl, M., Ardito, C. (eds.) International Conference on Web Engineering, pp. 49–65. Springer, Cham (2022)

[17] Cheng, G., Guo, L., Zhao, T., Han, J., Li, H., Fang, J.: Automatic landslide detection from remote-sensing imagery using a scene classification method based on bovw and plsa. IJRS **34**(1), 45–59 (2013)

[18] Ji, S., Yu, D., Shen, C., Li, W., Xu, Q.: Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks. Landslides **17**(6), 1337–1352 (2020)

[19] Tavakkoli Piralilou, S., Shahabi, H., Jarihani, B., Ghorbanzadeh, O., Blaschke, T., Gholamnia, K., Meena, S.R., Aryal, J.: Landslide detection using multi-scale image segmentation and different machine learning models in the higher himalayas. Remote Sensing **11**(21), 2575 (2019)

[20] Prakash, N., Manconi, A., Loew, S.: A new strategy to map landslides with a generalized convolutional neural network. Scientific reports **11**(1), 1–15 (2021)

[21] Hölbling, D., Füreder, P., Antolini, F., Cigna, F., Casagli, N., Lang, S.: A semi-automated object-based approach for landslide detection validated

by persistent scatterer interferometry measures and landslide inventories. Remote Sensing **4**(5), 1310–1336 (2012)

[22] Ju, Y., Xu, Q., Jin, S., Li, W., Su, Y., Dong, X., Guo, Q.: Loess landslide detection using object detection algorithms in northwest china. Remote Sensing **14**(5), 1182 (2022)

[23] Choi, C.E., Cui, Y., Zhou, G.G.: Utilizing crowdsourcing to enhance the mitigation and management of landslides. Landslides **15**(9), 1889–1899 (2018)

[24] Can, R., Kocaman, S., Gokceoglu, C.: A convolutional neural network architecture for auto-detection of landslide photographs to assess citizen science and volunteered geographic information data quality. ISPRS International Journal of Geo-Information **8**(7), 300 (2019)

[25] Can, R., Kocaman, S., Gokceoglu, C.: Development of a CitSci and artificial intelligence supported GIS platform for landslide data collection. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **43**, 43–50 (2020)

[26] Razis, G., Theofilou, G., Anagnostopoulos, I.: Latent twitter image information for social analytics. Information **12**(2), 49 (2021)

[27] Alfarrarjeh, A., Agrawal, S., Kim, S.H., Shahabi, C.: Geo-spatial multimedia sentiment analysis in disasters. In: DSAA, pp. 193–202 (2017)

[28] Weber, E., Marzo, N., Papadopoulos, D.P., Biswas, A., Lapedriza, A., Ofli, F., Imran, M., Torralba, A.: Detecting natural disasters, damage, and incidents in the wild. In: ECCV, pp. 331–350 (2020). Springer

[29] Imran, M., Alam, F., Qazi, U., Peterson, S., Ofli, F.: Rapid damage assessment using social media images by combining human and machine intelligence. In: ISCRAM, pp. 1–13 (2020)

[30] Ofli, F., Alam, F., Imran, M.: Analysis of social media data using multimodal deep learning for disaster response. In: ISCRAM, pp. 1–10 (2020)

[31] Musaev, A., Wang, D., Pu, C.: Litmus: Landslide detection by integrating multiple sources. In: ISCRAM (2014)

[32] Musaev, A., Wang, D., Xie, J., Pu, C.: Rex: Rapid ensemble classification system for landslide detection using social media. In: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), pp. 1240–1249 (2017). IEEE

[33] Fleiss, J.L.: Measuring nominal scale agreement among many raters. Psychological bulletin **76**(5), 378 (1971)

[34] Bayerl, P.S., Paul, K.I.: What determines inter-coder agreement in manual annotations? a meta-analytic investigation. Computational Linguistics **37**(4), 699–725 (2011)

[35] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. In: ICML, pp. 647–655 (2014). http://jmlr.org/proceedings/papers/v32/donahue14.html

[36] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. In: ICLR (2014)

[37] Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and transferring mid-level image representations using convolutional neural networks. In: CVPR, pp. 1717–1724 (2014)

[38] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)

[39] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90

[40] Huang, G., Liu, Z., Maaten, L.v.d., Weinberger, K.Q.: Densely Connected Convolutional Networks. In: CVPR, pp. 2261–2269 (2017). https://doi.org/10.1109/CVPR.2017.243

[41] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: CVPR, pp. 2818–2826 (2016)

[42] Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: ICML, pp. 6105–6114 (2019)

[43] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)

[44] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. IJCV **115**(3), 211–252 (2015)

[45] Chicco, D., Jurman, G.: The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. BMC genomics **21**(1), 1–13 (2020)

[46] Van der Maaten, L., Hinton, G.: Visualizing high-dimensional data using t-SNE. JMLR **9**(11), 2579–2605 (2008)

[47] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR, pp. 2921–2929 (2016)

[48] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021). https://openreview.net/forum?id=YicbFdNTTy

[49] Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 702–703 (2020)

[50] Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6023–6032 (2019)

[51] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research **15**(1), 1929–1958 (2014)

[52] Imran, M., Ofli, F., Caragea, D., Torralba, A.: Using AI and social media multimodal content for disaster response and management: Opportunities, challenges, and future directions. Elsevier (2020)