Predictive Artificial Neural Network For Assessing Sidewalk Accessibility Labels From Unlabeled Crowdsourced Data

Chu Li
Paul G. Allen School of Computer Science
University of Washington
chuchuli@cs.washington.edu

Zhihan Zhang
Paul G. Allen School of Computer Science
University of Washington
zzhihan@cs.washington.edu

Abstract

In recent work, machine learning methods have been applied to automatically identify pedestrian infrastructure in online map imagery. While promising, these methods have been limited by computational capacity and image availability. In light of the more than 700,000 labels present in the Project Sidewalk database, we propose the use of non-image label metadata to train neural networks to infer accessibility label accuracy. We apply automated labeling functions to unlabelled data, and develop an artificial neural network trained on noisy probabilistic training labels. Our approach yields precision results exceeding 90%, which further contributes to Project Sidewalk's overarching research agenda that is aimed at developing fast and accurate semi-automated sidewalk assessment tools that can be used to improve urban accessibility.

1 Introduction

Sidewalks form the backbone of cities: they can provide a safe, off-road pathway for pedestrians, support environmentally friendly mobility, and promote local commerce, recreation, and social interaction [1, 3]. For people with disabilities and older adults, sidewalks provide access to critical services and first/last mile transit. And yet, unlike their road counterparts, there is a lack of high-quality sidewalk datasets and fast, inexpensive, and reliable sidewalk assessment techniques [2, 6]—which fundamentally limits how we study and plan equitable urban infrastructure and mobility.

Project Sidewalk offers a scalable approach to accurately, efficiently, and cost-effectively locate and evaluate sidewalks through remote crowdsourcing and online map imagery. Project Sidewalk (https://projectsidewalk.org) is an open-source crowdsourcing platform that allows online users to label sidewalk conditions and identify accessibility issues through engaging missions and street scene imagery, similar to a first-person video game. For each sidewalk label, the platform collects information including the label type, a severity score of the problem, relevant tags, and optional descriptive text. Project Sidewalk uses gamified missions to train, engage, and sustain users and to divide tasks. Since its 2018 pilot deployment in DC, the project team has worked with partners and NGOs to deploy Project Sidewalk into 12 additional cities across North America, Europe and Asia [13, 5]. As of March 2023, a total number of 10,985 users contributed to 757,730 labels. We believe this is the largest open sidewalk accessibility dataset in existence.

In addition to *labeling missions*, Project Sidewalk introduces *validation missions* to counter the noisy nature of crowd-sourced data. In *validation missions*, users review and validate previously labeled imagery through agree, disagree, and unsure judgments. The problem with the current validation system is that the group of crowdworkers who are mislabeling the sidewalk features are also the crowdworkers who are validating other people's labels. For example, one typical mistake

Preprint. Under review.

is mislabeling driveways as curb ramps. While driveways are often used as a last-resort accessible pathways, they are not ADA regulated and should not be labeled as curb ramps. Despite this, many driveways mislabeled as curb ramps are validated as correct in the Project Sidewalk database. When such data noise occurs at scale, it can skew analyses and study results based on Project Sidewalk data.

Previous studies have employed image-based machine learning (ML) techniques to assess sidewalk accessibility features in Google Street View (GSV) images [14]. However, image-based approaches have significant limitations. Firstly, it can be expensive to train image-based models, particularly deep neural networks, which demand significant computational power and memory. Secondly, privacy concerns and updated imagery cause many GSV images to become unavailable, currently accounting for at least 30% of the Project Sidewalk database. In order to overcome the limitations of prior image-based approaches, we propose the following research question: **How can we utilize non-image label metadata to train ML models to predict label accuracy?**

We make the following key contributions:

1) For the first time, automated labeling functions that draw upon domain knowledge and heuristics from urban planning are used to label the unlabelled dataset. Subsequently, we employ Programmatic Weak Supervision (PWS) framework to generate noise-aware probabilistic training labels for performing supervised learning.

2) We develop and fine-tune a Multilayer Perceptron (MLP) classifier utilizing the labels generated by PWS to forecast the precision of sidewalk accessibility labels. We achieve a precision of over 90% for 4 out of 5 main label types.

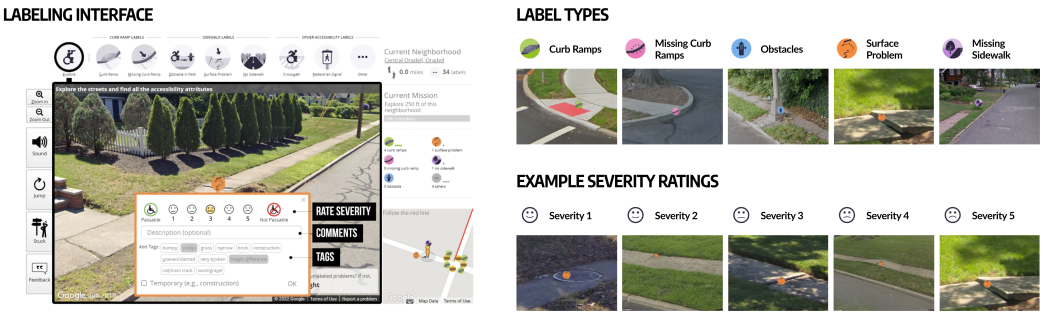

Figure 1: Project Sidewalk Labeling Interface, Label Types & Severity Rating Examples

2 Related Work

2.1 Project Sidewalk

Recent work has applied image-based deep learning models to automatically find and detect sidewalk accessibility problems for people with varying mobility, such as cracked pavement or overgrown vegetation, in online GSV panorama imagery [14]. The work presented a trained convolutional neural network (CNN) that can recognize patterns in the images that indicate the presence of accessibility problems, and has shown to significantly improve upon prior Support Vector Machine (SVM) based automated method [7], in some cases exceeding human labeling performance. However, neural networks are notably training-data-hungry. The noisy nature of Project Sidewalk's crowdsourced data undermines deep learning model training. Furthermore, the lack of publicly available datasets of labeled sidewalk accessibility images and the uncertain availability of GSV imagery make it more difficult to train and evaluate the proposed method. Finally, the method is based on ResNet, which makes it dependent on high-resolution images, further limiting its use in certain situations.

To improve the accuracy of the crowd-sourced dataset of Project Sidewalk, recently, Duan et al. [4] have studied a crowds plus machine learning (ML) technique to semi-automatically assess sidewalk accessibility features in GSV images. The study compared the positively validated data, i.e., data that has been voted correct by the crowd, with a larger but noisier aggregate dataset, and found that precision and accuracy for the specific types of accessibility problems increased, however the improvement was minimal. While the proposed assumptions are willing, questions remain about

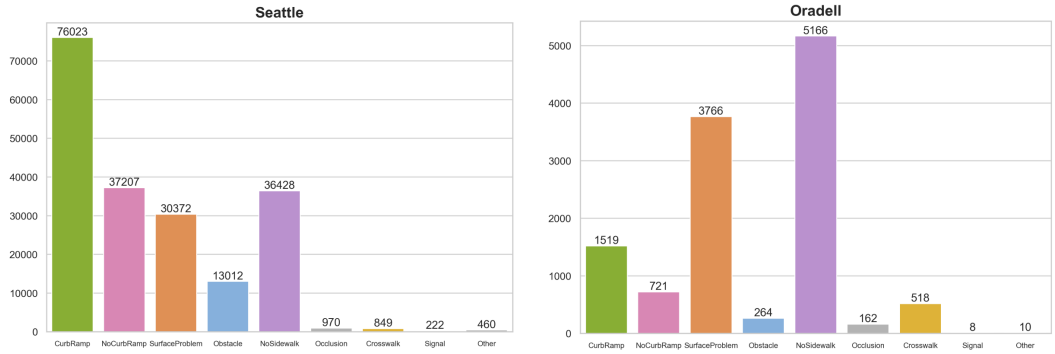

Figure 2: Training Dataset Distribution by Label Type and Severity

how to obtain accurate crowd-sourced datasets of sidewalk accessibility features in a cost-effective manner.

2.2 Neural Network with Limited Data

The non-image-based variables used in this study are mainly nominal (e.g., label type) and ordinal categorical data (e.g., severity rating, agreement count), making a Neural Network (NN) well-suited for this problem. However, the training of a NN is limited by the amount of clean training data available with the existing Project Sidework dataset. To address the data-hungry nature of training these NN models, the recently proposed Programmatic Weak Supervision (PWS) framework provided promising solutions [12, 11]. PWS is a method for training NN models with limited labeled data, it aggregates the noisy votes of labeling functions using a set of heuristics, e.g., keywords and domain knowledge, to produce training labels. These training labels are then used to pre-train a model for downstream tasks. PWS reveals the probability of addressing the challenge of limited labeled data for training a NN.

3 Data Collection

3.1 Project Sidewalk Data

The Project Sidewalk dataset comprises 757,730 labels, each of which is assigned to one of the following types: *curb ramps, missing curb ramps, sidewalk obstacles, surface problems, missing sidewalks, occlusion, crosswalk, signal* and *others*. Each label includes a severity assessment on a scale of 1 to 5, with 5 being the most severe, indicating a scenario that is impassable for a wheelchair user. Additionally, labels may include an optional open-ended description and one or more label-specific tags. All labels are accompanied by metadata, such as the date the GSV image was captured, the date and time the label was assigned, validation information, and geographical location (latitude and longitude).

For the purposes of this project, we use Project Sidewalk labels from Seattle and Oradell as our starting point. Our selection of these two locations is intentional, as they represent urban areas with distinct characteristics: Seattle being a major city and Oradell being a suburban locale. This deliberate choice would help us develop ML models that could effectively account for the diverse urban compositions. In total, Project Sidewalk provides 195,543 labels for Seattle and 12,134 labels for Oradell. Figure 2 illustrates the distribution of data by label type for the two cities. The Project Sidewalk research team validated 16,580 and 4,143 of these labels for Seattle and Oradell, respectively, which was used as our ground truth dataset.

3.2 Open Street Map Data

OpenStreetMap (OSM) (https://www.openstreetmap.org/) contains a wide variety of geographic data, including spatial information about roads, buildings, land use and topography. Our major use of

OSM was to obtain spatial information about roads and sidewalks, including hierarchy, geometry and location, in order to pair with our Project Sidewalk labels for subsequent spatial analysis.

3.3 Research Hypotheses

Drawing on the available datasets, distinct label characteristics, the nature of crowdsourcing, observations of user behavior in Project Sidewalk, and research in urban planning guidelines, we propose the following hypotheses:

Label Type. Project Sidewalk has 5 major lable types: *curb ramps, missing curb ramps, sidewalk obstacles, surface problems*, and *missing sidewalks*. Although most users can identify *missing sidewalks* with relative ease, accurately labeling other types of features may require careful review of the tutorial or prior knowledge. Hence, we anticipate that the accuracy of the model will vary based on each label type.

Severity Rating. Project Sidewalk's label severity ranges between 1-5, with 5 being the most severe, indicating a scenario that is impassable for a wheelchair user. While 1 and 5 are easy to judge, other severity levels tend to be more controversial.

Proximity. A label is more likely to be correct if it is placed closer to existing labels of the same type.

Optional input. When a user places a label, they will also be asked to add a description (comment) and relevant tags. For instance, *fire hydrant* and *pole* are the tags associated with the label type *obstacle*. Due to the fact that these input fields are optional, we expect that labels with such additional information will have a higher level of accuracy.

GSV zoom/ pitch/ heading. In most cases, changing the default parameters of GSV results in a more accurate label. For example, When a user zooms in to place a label, the label is probably correct.

Label location. The positioning of a label in relation to the sidewalk can serve as an indicator of its accuracy. *Curb ramps*, for instance, should only occur at road intersections. If they fall outside a certain radius of an intersection, they are likely to be incorrect. Similarly, if *no sidewalk* labels are placed in proximity with existing sidewalk geometry, they are high likely to be false.

Land use and zoning. Previous studies [9, 8] have shown that sidewalk label quality varies with land use. In the case of Seattle, label quality is higher in commercial areas, while people tend to mistake driveways for curb ramps in residential areas [9].

3.4 Data Processing

Our datasets were processed according to our research hypotheses, primarily using spatial processing techniques to measure the distance between labels and road/sidewalk geometry. We also employed spatial clustering to determine if a label was near other existing labels. The detailed processed label data can be found in Table 1. For the scope of this report, we will discuss only our spatial clustering method in detail.

Spatial clustering. In order to determine whether a label is in proximity to others, we adopt the two-step clustering approach employed in Project Sidewalk [13]: single-user clustering followed by multi-user clustering. Firstly, we merge the raw labels provided by each user into intermediate clusters, as some users may label a single issue from multiple angles. Secondly, we merge these user-specific clusters to form our final cluster dataset. Both steps utilize the Vorhees clustering algorithm along with the haversine formula to calculate distances between labels and clusters [13].

During the first step, we cluster raw labels of the same type that are within a designated distance threshold. As some label types may naturally be close together, such as two curb ramps on a corner, we use two different thresholds of 2 meters for curb and missing curb ramps and 7.5 meters for other label types. These thresholds were determined through previous Project Sidewalk empirical analysis, where clusters were calculated at different threshold levels ranging from 0 to 50 meters (with a step size of 1 meter) and evaluated qualitatively [13]. The second step of clustering is similar, but it uses the centroids of the first-step clusters with slightly broader thresholds of 7.5 and 10 meters, respectively [13].

category	column	type	note
label characteristics	label_id	int	unique label identifier
	user_id	str	unique user identifier
	label_type	str	one of the seven label types
	severity	int	severity value rated by the user
	gsv_panorama_id	str	unique identifier for the panorama
	geometry	point	cooordinates of the label in longtitiude and latitude
user behaviours	zoom	int	gsv zoom at the time of label placement
	heading	float	gsv heading at the time of label placement
	pitch	float	gsv pitch at the time of label placement
	photographer_heading	float	original gsv heading at the time of panorama taken
	photographer_pitch	float	original gsv pitch at the time of panorama taken
	tag_list	bool	whether or not the label has an associated tag
	tag_count	int	number of tags associated with the label
	description	bool	whether or not the label has an associated comment
crowdsource nature	clustered	bool	whether or not the label belong to a cluster
	cluster_count	int	the number of labels in the cluster
planning guidelines	distance	float	distance in ft to the closest sidewalk geometry
	way_type	str	road hierarchy label
	intersection_distance	float	distance in ft to the closest intersection

Table 1: Dataset Features Overview

4 Methods

We aim to predict the accuracy of Project Sidewalk accessibility labels, our general pipeline is presented in Figure 3. Our approach accomplishes the following four goals: 1) employing automated labeling functions, informed by domain knowledge and heuristics from urban planning, to label the unlabelled dataset, 2) employing PWS framework to generate a matrix of noise-aware probabilistic training labels for each data point for performing supervised learning, 3) constructing an MLP classifier to predict the precision of sidewalk accessibility labels utilizing the labels obtained from PWS, 4) fine-tuning the MLP model using a small, clean dataset to enhance the performance on specific label classification tasks.

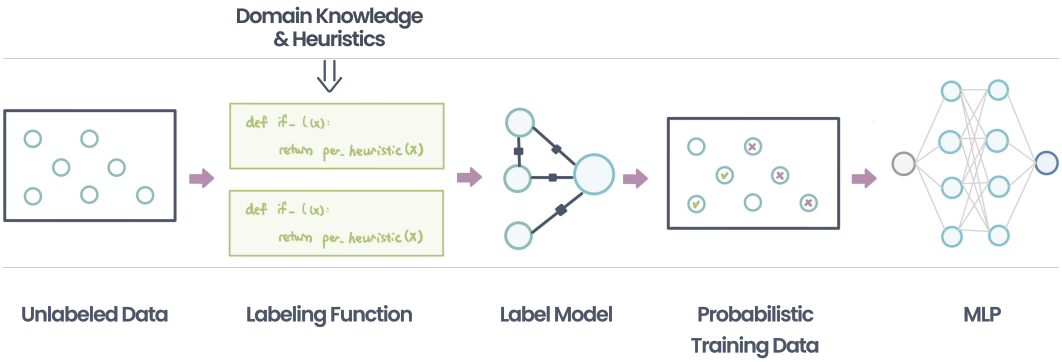

Figure 3: Schematic of PWS pipeline. Labeling functions, consisting of programmatic rules and heuristics, are employed to assign labels to the unlabeled training data. Subsequently, Label Model is trained to estimate the accuracy of each labeling function, producing a vector of probabilistic training labels for the training of MLP.

4.1 Weak Supervision

Instead of manually annotating our training data, we adopted an automated approach using labeling functions (LFs) to label our dataset. We employed the popular system, Snorkel [10], to establish a weak supervision pipeline. Snorkel allows for the integration of domain knowledge and heuristics into models and provides a method for estimating their accuracy and correlation in a consistent manner. As a result, the training labels can be reweighted and combined to create high-quality labels. We find this approach particularly suitable for our project for two reasons: first, it enables us to incorporate domain knowledge into our models, such as urban planning guidelines; second, it allows us to train on unlabelled data.

Assume there is a label matrix Λ, where $\Lambda_{i,j} = \lambda_j(x_i)$, we can encode the labeling functions using three factor types:

$$\phi_{i,j}^{label}(\Lambda, Y) = \mathbb{1}\{\Lambda_{i,j} \neq \varnothing\}$$

$$\phi_{i,j}^{accuracy}(\Lambda, Y) = \mathbb{1}\{\Lambda_{i,j} = y_i\}$$

$$\phi_{i,j,k}^{correlation}(\Lambda, Y) = \mathbb{1}\{\Lambda_{i,j} = \Lambda_{i,k}\}$$

See below for a set of 7 labeling algorithms we derived from our research hypothesis. For example, Algorithm 1 is predicated on the observation that users may inadvertently mislabel driveways as curb ramps in residential areas. According to planning guidelines, *curb ramps* are typically installed at intersections only. Thus, Algorithm 1 indicates that if a *curb ramp* or *missing curb ramp* label in a residential area falls outside a certain radius from the intersection, the label is likely to be false.

Algorithm 1 Intersection

if label $\in$ CurbRamp | NoCurbRamp$\wedge$label $\in$ residential $\wedge$ label $\notin \forall$road intersection **then**
    label = false
else
    $\Diamond$label = false
end if

Algorithm 2 Cluster

if $\sum(\exists$labels $\Rightarrow \Delta(distance) < 10m) > 4$ **then**
    labels = true
else
    $\Diamond$labels = true
end if

Algorithm 3 Zoom

if zoom > 2 where zoom $\in$ label **then**
    label = true
else
    $\Diamond$label = true
end if

Algorithm 4 Severity

if severity ≥ 4 where severity $\in$ label **then**
    label = true
else
    $\Diamond$label = true
end if

Algorithm 5 Tag

if $\exists$ tag $\in$ label **then**
    label = true
else
    $\Diamond$label = true
end if

Algorithm 6 Description

if $\exists$ description $\in$ label **then**
    label = true
else
    $\Diamond$label = true
end if

Algorithm 7 Sidewalk Distance

if label $\in$ NoSidewalk $\wedge$ label $\in \forall$sidewalk **then**
    label = false
else if label $\in$ Obstacle | SurfaceProblem $\wedge$ label $\notin \forall$sidewalk **then**
    label = false
else
    $\Diamond$label = false
end if

4.2 Label Model

The labeling functions used in this process are prone to noise and inaccuracies, and may overlap with one another. The LFs abstractly provide a flexible interface label function abstraction. To address these challenges, we applied a Label Model that is capable of denoising the signals and reducing the need for manual tuning.

The Label Model takes the full set of labeling functions as input and applies them using the LFApplier to obtain label matrices. However, it's important to note that labeling functions have different properties and should not be treated equally. In addition to varying accuracy and coverage, labeling functions may be correlated, leading to the overrepresentation of certain signals in a simple majority-vote-based model. To handle these complexities, we employed a more sophisticated model to combine the outputs of the labeling functions.

LFs label instances independently, assuming knowledge of the true class label. Each labeling function has a certain probability of labeling an instance and a probability of correctly labeling it. To maximize the probability of the observed labels produced on our training examples occurring under the generative model, we use Stochastic Gradient Descent to optimize the solution.

This model generates a single set of noise-aware, probabilistic training labels, which will be used to train a Neural Network(NN) classifier for our task. The trained model h_θ utilizes our probabilistic labels $\tilde{Y}$ through the minimization of a noise-aware variant of the loss function $L(h_\theta(x_i), y)$, which computes the expected loss relative to Y:

$$\hat{\theta} = \arg\min_{\theta} \sum_{i=1}^{m} \mathbb{E}_{y\sim Y}[L(h_\theta(x_i), y)]$$

4.3 Baseline

To establish a baseline for our analysis, we create a random forest model, which is built upon the high-dependency features selected through the methods outlined in the previous sections.

The balanced ground-truth dataset is randomly split for training and testing. Parameters are tuned based on the average accuracy of the testing results. We conducted a grid search over three key parameters: The maximum depth of the tree (search range: 2-20, with a step size of 1); the minimum number of samples required to split (search range: 2-500, with a step size of 10); the maximum number of leaf nodes (search range: 2-20, with a step size of 1). We determined the optimal values for the depth of the tree of 3, the minimum number of samples required to split of 182, and the maximum number of leaf nodes of 8. Finally, we trained a random forest, and selected example trees are visualized in Figure 4. By using random forest, we aim to identify the most important predictors for our problem and gain insights into the relationships between the features and the target variable.

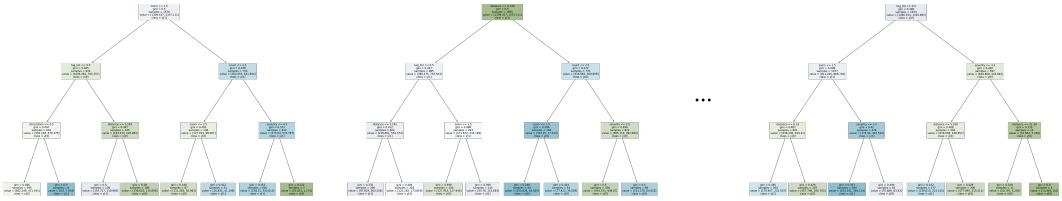

Figure 4: Visualization of random forest. In each node of trees, the "value" indicates the size of two group types. The blue color indicates the correct label group (class = y[1]), while the green color indicates the false label group (class = y[0]).

4.4 Multilayer Perceptron

The probabilistic training labels generated by our PWS are well-suited for training complex NN models that require a significant amount of data. Given that our dataset primarily consists of categorical features, we have decided to employ the weak dataset from PWS as training data for a

Multilayer Perceptron (MLP) classifier, which is particularly effective when dealing with nonlinearly separable input variables that exhibit uncertain dependencies.

Our approach involves constructing an MLP with four layers, consisting of input, output, and two hidden layers, with the number of neurons in the first and second hidden layers set at eight and four, respectively. Rectified Linear Unit (ReLU) is used in the hidden layers as activation functions to enhance convergence. The input neurons in the MLP correspond to the selected features from the dataset, with the probabilistic labels serving as neuron weights, providing a more comprehensive understanding of the problem.

For the output layer, we chose the sigmoid activation function to calculate the output probabilities in the range [0,1], which is commonly used for binary classifications. We selected sigmoid over softmax due to its faster computation time during backpropagation. This is because it doesn't require computing an exponential term and normalization for each neuron in the output layer. A schematic illustration of our MLP network architecture is shown in Figure 5.

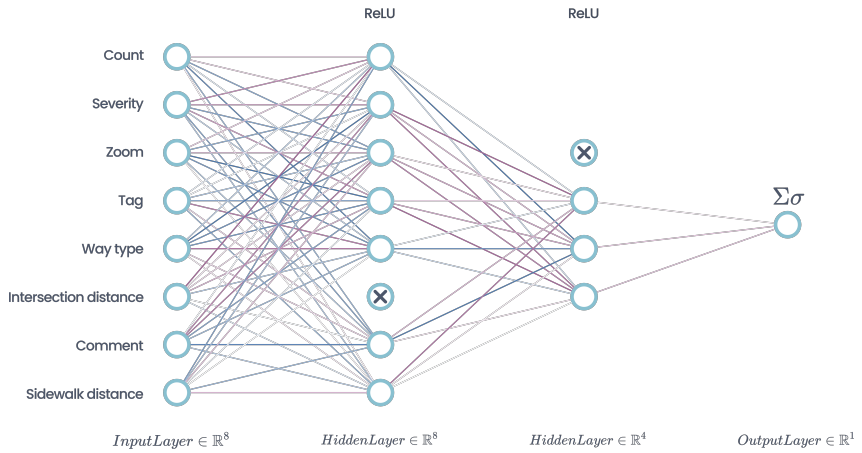

Figure 5: The MLP architecture for assessing sidewalk accessibility labels. Input is initially normalized; multiple hidden layers with ReLU are then applied, where each is followed by a dropout layer; finally, sigmoid activation function is used for the output.

To prevent overfitting, we also incorporated dropout, a regularization technique that randomly drops out a percentage of the neurons during training, reducing their co-dependency and improving the model's generalization performance. The dropout rate of 0.2 is chosen to strike a balance between reducing overfitting and maintaining an adequate level of model complexity.

4.5 Fine-tuning

Training a neural network on a small dataset can lead to overfitting, which can significantly impair the model's ability to generalize. To address this issue, we propose pre-training the network on a larger dataset, such as the complete Seattle Project Sidewalk dataset. The pre-trained model with learned relevant features from the comprehensive, but noisy dataset, can be further optimized to enhance the performance for each label-type classification tasks by fine-tuning on a smaller dataset, that only contains one specific label type. This is because the features of each label type are not significantly distinct, however, each label type contains distinct chracteristics, that can be further utilized to improve the overall performance of the model. A schematic illustration of our fine-tuning topology is shown in Figure 6.

5 Results

5.1 Evaluation Criteria

Precision was selected as the primary evaluation metric for our models, as it is a key evaluation metric for binary classification tasks. In our specific task, false positive predictions can have significant

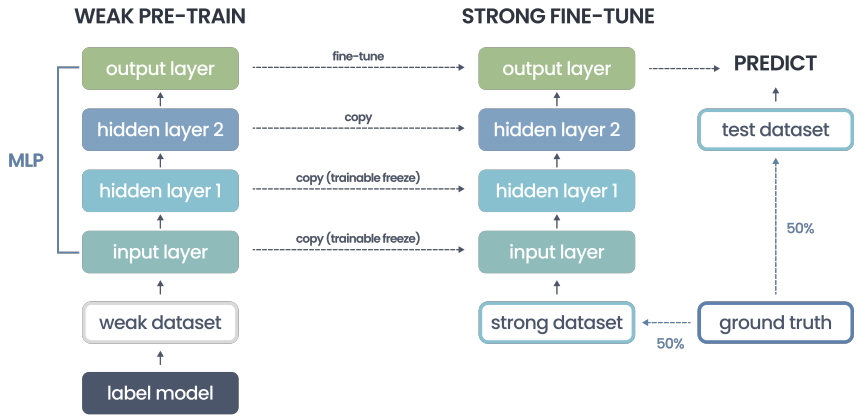

Figure 6: Fine-tuning topology. The pre-trained model on a large dataset is fine-tuned using a smaller dataset for the specific label type, with only the last hidden layer and output layer being trainable.

consequences and incur high costs. For example, a false positive result of a label validation prediction will cause the wrong label not being sent back to the research team for the validation process, which may further mislead the crowdsource labeler in the future. For instance, if a label validation prediction yields a false positive result, the incorrect label would not be sent back to the research team for revalidation, potentially leading to confusion and errors for the crowdsourced labelers in the future, which could ultimately undermine the quality and reliability of our dataset. Given this potential impact, the cost of false positives is higher than that of false negatives in our application.

Furthermore, our testing dataset is imbalanced, with significantly more positive instances than negative ones. In such cases, precision can provide a more accurate evaluation of the model's performance than accuracy, as accuracy can be biased towards the majority class.

5.2 Comparison Between Baseline & Label Model

The results of the Random Forest showed that the *label_tag* feature resulted in a Gini impurity of 0.115. We leveraged the inherent topology of the random forest, i.e., the node splitting conditions, to tune the parameters of our labeling functions.

Below we present the label model, pre-training and fine-tuning MLP results in the full Seattle Project Sidewalk dataset. For fair comparisons, we split the ground truth dataset equally into fine-tuning and testing sets (50/50) using the same random state. This ensures that we have a comparable model capacity and a consistent test dataset with the random forest baselines.

For fine-tuning, we freeze all layers except for the last hidden layer and output layer. We also reduce the learning rate to prevent overfitting.

Results are presented in Table 2, which show that MLP is competitive with the label model in terms of precision. However, fine-tuning MLP significantly improves performance in all criteria, as expected.

Model	Curb Ramp	Missing Curb Ramp	No Sidewalk	Surface Problem	Obstacle
Random Forest	93.0%	90.8%	82.8%	88.3%	62.3%
LM	92.7%	**91.5%**	**95.5%**	90.0%	64.9%
MLP	91.9%	89.5%	84.6%	91.2%	67.7%
MLP + Fine-tuning	**97.7%**	90.4%	92.5%	**92.4%**	**69.3%**

Table 2: Precision Score Comparison

5.3 Extending To Other Cities

Our study employed an identical model to analyze the Oradell dataset, and the obtained results are presented in Table 3. As indicated below, the precision score for the *obstacle* category was considerably lower for Seattle (69.3%), whereas it significantly increased for Oradell (92%). Upon conducting tag analysis, it was revealed that the complexity of obstacle situations in Seattle may have contributed to the observed discrepancies. Specifically, the tag information in Oradell primarily associated obstacles with trees/vegetation (40%), whereas in Seattle, obstacles were tagged with poles, trash/recycling can, vegetation, and parked cars in similar frequencies of around 20%. These findings highlight that the performance of our model for *obstacle* labels is superior in simpler situations, and further refinement is required to accommodate the complexities of urban environments.

City	Curb Ramp	Missing Curb Ramp	No Sidewalk	Surface Problem	Obstacle
Seattle	97.7%	90.4%	92.5%	92.4%	69.3%
Oradell	97.4%	72.6%	96.8%	97.4%	92.0%

Table 3: Precision Score per Label Type per City

6 Discussion

6.1 Datasets

Extending to additional cities. The current iteration of our algorithm has only been trialed in the city of Seattle, WA and Oradell, NJ. Our aim is to extend its application to 11 additional cities featured in the Project Sidewalk database.

6.2 Modeling

Landuse & zoning. One of our original research hypotheses included the use of land use and zoning information to predict label precision, but this hypothesis has not yet been implemented. Previous studies [9, 8] have shown that sidewalk label quality varies with land use. In the case of Seattle, label quality is higher in commercial areas, while people tend to mistake driveways for curb ramps in residential areas [9]. In the next stage, we plan to incorporate this heuristic into our labeling functions.

Relationships between different label types. Taking into account the interdependence between different label types may help in developing more accurate models. Specifically, certain labels such as *surface problems* and *obstacles* are contingent upon the presence of sidewalks. Consequently, labels that indicate such issues in areas where *missing sidewalk* labels are present are likely to be inaccurate. Similarly, *curb ramp* and *missing curb ramp* labels are mutually exclusive. Therefore, considering such relationships may improve the precision and reliability of model predictions.

6.3 Applications

Human-AI interaction The outcomes of this project have practical implications for the design of human-AI interaction in the Project Sidewalk platform. Specifically, the model can enhance crowdsourced data precision by providing timely feedback when inputs are predicted to be inaccurate. Suppose a user identifies a curb ramp and the model predicts it is likely to be false, then a message will appear asking, "Are you sure this is a curb ramp?" As next steps, we plan to test its effectiveness with minimally-trained crowdworkers. We hope this will improve the quality of the Project Sidewalk data and contribute to other efforts in urban accessibility research.

Acknowlegments

We would like to thank Prof. Tim Althoff and Prof. Jon Froelich for their invaluable guidance throughout this project. We would also like to extend our special thanks to our TA Esteban Safranchik, whose contributions have been instrumental to the outcome of this project.

References

[1] S. A. Carlson, J. E. Fulton, M. Pratt, Z. Yang, and E. K. Adams. Inadequate Physical Activity and Health Care Expenditures in the United States. *Progress in Cardiovascular Diseases*, 57(4):315–323, Jan. 2015.

[2] S. Deitz, A. Lobben, and A. Alferez. Squeaky wheels: Missing data, disability, and power in the smart city. *Big Data & Society*, 8(2):20539517211047735, July 2021. Publisher: SAGE Publications Ltd.

[3] D. Ding, T. Kolbe-Alexander, B. Nguyen, P. T. Katzmarzyk, M. Pratt, and K. D. Lawson. The economic burden of physical inactivity: a systematic review and critical appraisal. *British Journal of Sports Medicine*, 51(19):1392–1409, Oct. 2017. Publisher: BMJ Publishing Group Ltd and British Association of Sport and Exercise Medicine Section: Review.

[4] M. Duan, S. Kiami, L. Milandin, J. Kuang, M. Saugstad, M. Hosseini, and J. E. Froehlich. Scaling Crowd+AI Sidewalk Accessibility Assessments: Initial Experiments Examining Label Quality and Cross-city Training on Performance. In *The 24th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–5, Athens Greece, Oct. 2022. ACM.

[5] M. Duan, A. Kumar, M. Saugstad, A. Zeng, I. Savin, and J. E. Froehlich. Sidewalk Gallery: An Interactive, Filterable Image Gallery of Over 500,000 Sidewalk Accessibility Problems. In *The 23rd International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–5, Virtual Event USA, Oct. 2021. ACM.

[6] Y. Eisenberg, A. Heider, R. Gould, and R. Jones. Are communities in the United States planning for pedestrians with disabilities? Findings from a systematic evaluation of local government barrier removal plans. *Cities*, 102:102720, July 2020.

[7] K. Hara, J. Sun, R. Moore, D. Jacobs, and J. Froehlich. Tohme: detecting curb ramps in google street view using crowdsourcing, computer vision, and machine learning. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, UIST '14, pages 189–204, New York, NY, USA, Oct. 2014. Association for Computing Machinery.

[8] M. Hosseini, M. Saugstad, F. Miranda, A. Sevtsuk, C. T. Silva, and J. E. Froehlich. Towards Global-Scale Crowd+AI Techniques to Map and Assess Sidewalks for People with Disabilities, Aug. 2022. arXiv:2206.13677 [cs].

[9] C. Li, L. Orii, M. Saugstad, S. J. Mooney, Y. Eisenberg, D. Labbé, J. Hammel, and J. E. Froehlich. A Pilot Study of Sidewalk Equity in Seattle Using Crowdsourced Sidewalk Assessment Data, Oct. 2022. arXiv:2211.11545 [physics].

[10] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Ré. Snorkel: rapid training data creation with weak supervision. *Proceedings of the VLDB Endowment*, 11(3):269–282, Nov. 2017.

[11] A. Ratner, B. Hancock, J. Dunnmon, F. Sala, S. Pandey, and C. Ré. Training Complex Models with Multi-Task Weak Supervision. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):4763–4771, July 2019. Number: 01.

[12] A. J. Ratner, C. M. De Sa, S. Wu, D. Selsam, and C. Ré. Data Programming: Creating Large Training Sets, Quickly. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[13] M. Saha, M. Saugstad, H. T. Maddali, A. Zeng, R. Holland, S. Bower, A. Dash, S. Chen, A. Li, K. Hara, and J. Froehlich. Project Sidewalk: A Web-based Crowdsourcing Tool for Collecting Sidewalk Accessibility Data At Scale. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–14, Glasgow Scotland Uk, May 2019. ACM.

[14] G. Weld, E. Jang, A. Li, A. Zeng, K. Heimerl, and J. E. Froehlich. Deep Learning for Automatically Detecting Sidewalk Accessibility Problems Using Streetscape Imagery. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*, pages 196–209, Pittsburgh PA USA, Oct. 2019. ACM.