Proceedings Paper:

Barrows, J., Radu, V. orcid.org/0000-0003-3502-4355, Hill, M. et al. (1 more author) (2022) Active learning with data distribution shift detection for updating localization systems. In: Proceedings of 2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN). 2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 29 Nov - 02 Dec 2021, Lloret de Mar, Spain. IEEE . ISBN 9781665447348

https://doi.org/10.1109/IPIN51156.2021.9662543

2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 29 Nov. - 2 Dec. 2021, Lloret de Mar, Spain

Active Learning with Data Distribution Shift Detection for Updating Localization Systems

Josh Barrows
University of Sheffield
jbarrows1@sheffield.ac.uk

Valentin Radu
University of Sheffield
v.radu@sheffield.ac.uk

Matthew Hill
Dstl, UK
mhill2@dstl.gov.uk

Fabio Ciravegna
University of Sheffield
f.ciravegna@sheffield.ac.uk

***Abstract*—Maintaining smartphone indoor localization systems operational and accurate is challenging. Landmark and WiFi based systems rely on up-to-date reference points and radio maps. These are expensive to refresh for most current systems because of the exhaustive site surveys these rely on. Any physical change in the environment may cause unpredictable transformation in the radio propagation medium or it may affect the typical navigation paths inside the building. Active Learning (AL) involves human intervention in annotating sensor samples collected from the target environment. However, the burden on users to perform this manual task should be minimized to make the approach more user-friendly. We propose a solution that improves AL by involving the users only when sensor data deviates from what is expected at a location. Our solution analyses the confidence of location estimations in the environment. We propose a robust data distribution shift detector that looks at several levels in our location estimation neural network to determine the confidence of predictions. During the training stage, the model builds high confidence for estimations in the areas where data is available. We show that data from outside the training distribution is detected as anomaly, which triggers the AL to prompt the user for her current location as label. These annotated samples and other augmented data are then used to update the location estimation model.**

***Index Terms*—indoor localization, active learning, data distribution shift, estimation confidence**

I. Introduction

A growing number of mobile applications and services rely on indoor localization for their effective operation. Indoor localization systems estimating the user location based on data from smartphone sensors have been explored for almost two decades with many successful examples and alternative implementations (see Section II). One approach is to train a location estimation model using training data collected from a target region [1], [2]. Elements such as Wi-Fi fingerprints, magnetic field and landmarks (doors, corners, etc.) are often used as references by indoor positioning systems based on a preliminary exhaustive data collection campaign (site survey).

However, these site surveys are expensive to perform. They require deliberate intervention from a specialist to explore the environment and collect the training data. Any small change in that indoor space alters the radio propagation medium, modifying received Wi-Fi fingerprints, and might interfere with the magnetic flux in a small region (caused by relocating powered equipment with an electromagnetic field). To calibrate for such changes, new site surveys are needed periodically.

More recent work rely on crowds of people to source those alterations from the target environment [1], [2]. Solutions such as Active Learning (AL) aim to involve the users in labeling the data (providing the ground-truth location), which is then used to retrain the model (or update radio maps). However, it is hard to assess when to prompt the user for her location. Too many notifications and the user may become annoyed with the system.

We propose to improve AL by knowing when to prompt the user for interventions based on data distribution shift detection. Sensor samples collected by the smartphone at run-time that are not in the data distribution of the current training dataset will be seen as anomaly by the system and prompt the user for their labels.

Data that is annotated by users with the ground truth location extends our training set. During the training, we force the neural network based location estimation model to learn the samples from the training set in order to identify them at run-time with high confidence. We make this training robust by introducing data augmentation to assist when training samples are not evaluated with high confidence. This data augmentation increases the number of samples available for training in those regions with very few training samples. Effectively, this process has the role of compensating data skewness and boosting estimation confidence uniformly across the sampled spaces. The model prediction confidence is determined by observing the confidence at several levels in the neural network through parallel Softmax layers associated to the main layers of the position estimation network (Section IV discusses the confidence estimation process in detail).

We perform the evaluation of the proposed data distribution shift detection on Wi-Fi fingerprint data. Then we assess the impact of this distribution shift detector in an AL setup. Any new samples that are detected within the building are analysed and only used to update an indoor localisation model if they are determined to be data shift samples, i.e. from a region of the building that has not been adequately mapped, because there are few, if any, similar samples in the available training set. We observe that our approach for AL accelerates the training of a robust location estimation model using very little additional samples, which lowers the burden on the users to annotate the data. The evaluation is presented in more details in Section V.

This paper makes the following contributions:

- We propose a data distribution shift detection solution for assessing the confidence of location prediction with a neural network taking Wi-Fi fingerprints as input.
- We train the location estimation model with data augmentation to boost its confidence for edge cases, such that any low estimations at run-time reflect regions that have never been explored.
- We incorporated our data distribution shift detector in an active learning system to reduce the number of samples that need user input to update the radio map of a building.

II. Background and Related Work

A. Indoor Localisation

Indoor localisation refers to pinpointing a user's position within a building by utilising an indoor positioning system. Several localisation systems exist today.

Wi-Fi fingerprinting is an indoor localisation method that uses location fingerprinting to empirically measure signal strengths for inferring locations [3]. This method typically involves two phases - the offline phase (performed before the system is operational), in which the received signal strengths are collected at specified locations throughout the target indoor region. The signal strength values of Access Points are saved in a vector and associated a 'ground-truth' point (the x and y coordinates of the sample location, and z for floor or altitude) as such: (x, y, z). These *(location, signal strength)* pairs are known as *fingerprints* and saved in a database, which can be processed to produce a *radio map*.

After deployment, there is the online phase, which uses the radio map to determine a location for the measurements of a mobile device. Periodic Wi-Fi scans from a mobile device create the run-time observations in a vector of signal strengths (fingerprint). This online fingerprint is compared to the radio map of past fingerprints to return a location estimate based on the best matching signal strength vector. A similar process is employed by magnetic fingerprint based systems and systems that rely on physical landmarks in the building (such as corners, doors, trajectory matching, etc.).

Many previous work use sensor based landmarks and Wi-Fi signatures for performing indoor positioning. 'SpotFi' [4] achieves "decimetre level localisation" (i.e. 10cm). This exemplifies the principles underlying the majority of indoor localisation systems: deployability, universality and reliability – the system runs on existing Wi-Fi installations, the system is able to localise any device with a Wi-Fi chip without any further hardware requirement, and finally, accuracy is key for the adoption of such systems for everyday usage.

Using Deep Neural Networks (DNN) to deal with the variant and unpredictable nature of radio signals is also an active research area. This approach requires a substantial amount of training data, but this is becoming more widely available through community repositories [5]. In [6], a four-layer DNN model extracts features from widely fluctuating Wi-Fi samples. This model is pre-trained as a Stacked Denoising Autoencoder (SDA), which learns reliable features automatically from datasets of noisy samples. In addition to the isolated Wi-Fi based position estimations, in [6] the trajectory is refined by a Hidden Markov Model (HMM), which smooths any occasional erroneous estimations.

A major problem for fingerprint based systems is the staleness of radio maps. Constant update is needed to cope with any involuntary and occurring changes in the environment. Research focused on algorithmic strategies for adapting to environmental changes in Wi-Fi location fingerprinting has been developed in recent years. In [3], the focus is to perform continuous updates to an existing radio map. Primarily, this is motivated by the characteristics of the propagation environment, with gradual attenuation of the radio signals, but also being affected by reflection, refraction, and absorption caused by building structures and people moving in the environment, resulting in significant distortions to the radio waves as it is interpreted at the receiving end. In their system, 'Streamspin', end users can choose to contribute their indoor location, whereby they can indicate they are at a previously profiled location or at a new location to expand the coverage of the radio map. This human contribution helps to maintain fresh radio maps.

Other systems replace the human intervention with infrastructure mounted cameras for position estimation [7]. In [8], a system tracks the movement of people to estimate their exact location when in view of the camera. Such context information from the vision component can automatically annotate sensor data [9] and continuously refine the estimation models.

In [10] AL in the context of Wi-Fi localisation is used for reducing the amount of labelled data required for training a Wi-Fi fingerprint model. This is designed to tackle the main barrier of broad adoption, the labour-intensive process of collecting labelled fingerprints. They show how a fingerprinting system can be constructed with noticeably less labels, while obtaining high positioning accuracy. Rather than requiring the user to stop at every location within the target building to annotate collected Wi-Fi fingerprints, dead reckoning is employed to predict subsequent locations and label the Wi-Fi fingerprints with. The accumulating error of dead reckoning is addressed by automatically identifying some strategic locations to label, and with AL the annotator is asked for her exact position when close to those strategic locations.

B. Active Learning

Active Learning (AL) is a subsection of machine learning. There are countless applications for this technology, including but not limited to speech recognition [11], textual information extraction [12] and image classification [13].

In [14], Burr Settles et al. explain the basic idea behind AL as being a machine learning method of learning which can reach greater accuracy by using fewer labelled training instances, if it is allowed to select which instances to learn from. Essentially, AL algorithms raise a *query* that requests a human annotator, known as an *oracle*, to label a previously unlabelled sample of data.

In [11], AL is used for reducing the examples to be labelled for training, by inspecting the unlabelled examples, and intelligently deciding the most informative instances with consideration for a cost function, such as the cost for the oracle to label. Their aim is to choose the data samples that will offer the best improvements in performance of the speech recogniser. AL offers information extraction strategies [12], including confidence and distance based approaches. The basic premise is to design an algorithm that automatically identifies documents for the user to annotate.

AL can also be used for novel image classification [13], by building a robust classifier from a limited amount of labelled training instances. They use this in an incremental learning manner. The unlabelled data is progressively fed into the Convolutional Neural Network (CNN). The majority of these samples are clearly classified and take the labels provided by the CNN. A minority of samples are user-annotated if they do not meet the required threshold level, both of which are then used to further update the CNN model with an increased training set.

To determine the uncertain samples for annotation, three AL approaches are commonly used: *least confidence (LC)*, *margin sampling (MS)*, and *entropy (EN)*. Their effect is to select the most informative/uncertain instances from the unlabelled set for annotation:

1) LC ranks the unlabelled samples in ascending order based on the probability of the sample belonging to a category (label). When a low probability for the sample belonging to its most probable class occurs, the classifier is uncertain about that sample and added to the ranking for annotation.
2) MS ranks the unlabelled samples based on the difference between the first and the second most probable class the classifier considers it belongs to. When there is a small margin between these top two probabilities, the classifier is uncertain about the sample so it needs to be manually annotated.
3) EN ranks the unlabelled samples on the difference between all class probabilities. A high value of entropy means an increased uncertainty about the sample's class.

The techniques are focused on enlarging the training dataset and using continuous data learning to retrain the classifier and adapt the model for new samples. The problem with this is that when learning new samples, old data points can become forgotten. This aspect is the focus of another research [15] that aims to alleviate the forgetting over time.

These are just a few examples of how AL can be applied to certain scenarios, however the underlying techniques are always similar, identifying the most relevant samples to be labelled for training.

C. On Smartphone Premises

The requirement for any AL solution to run within the computational confinements of a smartphone is of paramount importance. Smartphones are, however, the ideal device to use for indoor positioning, due to their wide array of sensors, and their ubiquitousness in society.

In order to fulfill the requirement of a *smartphone based* AL algorithm for indoor localisation, any final strategies or techniques deployed must be capable of running on a mobile device.

Updating location estimation models locally on a mobile device has its challenges. Mobile devices are constrained by their relatively limited processing capabilities (as opposed to a dedicated computer system, with greater resources), hence by using an opportunistic solution such as AL, only the most relevant sensor samples will be selected for model update, saving energy in communication and storage. This problem can therefore be managed by using efficient AL strategies.

Research into minimising the resource usage of neural networks [16] acknowledges that state-of-the-art uncertainty estimation methods have a high impact on the computational resources of constrained devices. Their approach allows pre-trained DNN models to produce uncertainty estimation for a classification task, being optimised for use on resource-limited devices. The efficient framework that directly enables these models to generate uncertainty estimates does not require any additional training or fine-tuning of the DNN model. The paper proposes that a layer-wise distribution is propagated through the network in a cascaded manner, massively reducing the computational complexity by allowing the model to produce uncertainty estimations in one single run. This approach makes it possible therefore to estimate the predictive uncertainty on a wider range of small devices, which would otherwise be harder with traditional techniques. Traditional methods need multiple runs to generate the prediction certainty [17].

III. Active Learning with Data Distribution Shift

A. Data Distribution Shift

A prediction system uses training (past) data covariate response pairs $\{x_{train}, y_{train}\}$ from a distribution $p(X, Y) = p(Y|X)p(X)$, having the conditional model $p(Y|X)$ and with a prior $p(X)$. This learns to estimate the condition $\hat{p}(Y|X)$ (predictor), such that it makes estimations for future variables from a test set $\hat{p}(y_{test}|x_{test})$.

Covariate shift occurs when the distributions of covariates are different between the training data and the test data [18]. This has implications when the true model $P(Y|X)$ cannot be approximated correctly by the prediction model $\hat{p}(Y|X)$. It could also be that the prediction model works well for $p(X_{train})$ but not for $p(X_{test})$, if the model is only accurate in the X-space.

The use of novelty detection models [19], is a method of learning a decision boundary between multiple categories in a dataset, allowing any novel classes to be detected during the testing phase. Further research [20] into data distribution shift detection analyse machine learning systems that fail loudly, built on solutions for detecting dataset shift. These identify exemplars that are obvious candidates for the shift, and quantifies shift malignancy (the amount of deviation). The findings show that precise subtle changes in the data distribution can degrade the performance of even state-of-the-art classifiers. As an extension, anomaly detection exposes data instances that deviate from the main density of data instances in a set [21]. A method of particular note is using SoftMax likelihood models. The aim of this approach is to generate a score that reflects the likelihood of events given an input. The training dataset will tune the model to produce maximum likelihood, or strong predictive confidences. The weaker confidence samples generally come from a different distribution, highlighting the anomalies in data. The concept of producing anomaly scores by modelling the event likelihood is expressed by:

$$\Theta^* = \arg\max_{\Theta} \sum_{\mathbf{x}\in\mathcal{X}} \log p(\mathbf{x};\Theta) \tag{1}$$

where $p(\mathbf{x};\Theta)$ is the probability of a given instance x, with Θ being the parameter to be learned. As so, $p(\mathbf{x};\Theta)$ is modelled with the SoftMax function:

$$p(\mathbf{x};\Theta) = \frac{\exp(\tau(\mathbf{x};\Theta))}{\sum_{\mathbf{x}\in\mathcal{X}} \exp(\tau(\mathbf{x};\Theta))} \tag{2}$$

where $\tau(\mathbf{x};\Theta)$ is the anomaly scoring function capturing the feature interactions:

$$\tau(\mathbf{x};\Theta) = \sum_{i,j\in\{1,2,\cdots,K\}} w_{ij}\mathbf{z}_i\mathbf{z}_j \tag{3}$$

where z_i is the embedding in a lower dimensionality for the i-th feature value of x in the representation space. W_{ij} is a trainable parameter representing the weight brought to the interaction.

In [21], SoftMax models are analysed based on their advantages, showing that anomaly detection scores using SoftMax can incorporate different types of interactions. They also come with disadvantages, most notably being the computation cost when the number of features in each data instance is large. This is due to a $O(D^n)$ time complexity per instance for the n-th order interactions of D features. However, given that most datasets used for indoor localisation are commonly very small, performance issues are not a key concern, and hence SoftMax likelihood models are a viable method for detecting anomalous samples as they offer an adequate approach to learn low-dimensional representations from a given training set. Other reviews [22] that evaluate out-of-distribution sample detectors also thoroughly investigate the usage of SoftMax in several detection methods, but have concluded that reliably evaluating the performance of out-of-distribution detectors is difficult [23]. Previous methods are ineffective and perform incomplete evaluations.

SoftMax layers are also shown to be efficient confidence estimators in deep neural networks [24]. Lower probability values in the SoftMax expose anomalous examples, so a classifier conveniently acts as a efficient out-of-distribution detector. Correctly classified examples produce greater values in the SoftMax than erroneously classified and out-of-distribution examples. In another research [24], three domains are investigated for anomaly detection with SoftMax: natural language processing, computer vision and automatic speech recognition. They show how the SoftMax exposes data that is from a different distribution than the training data distribution, in all three domains. We demonstrate that SoftMax as a confidence assessment mechanism is also applicable to classifying Wi-Fi samples and their distribution.

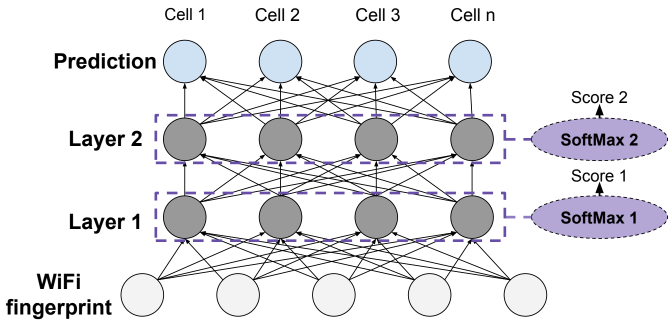

Fig. 1: Location estimation model with confidence assessment. Each layer of the model has a SoftMax associated in parallel to the main dataflow. SoftMax provides the largest probability to be aggregated in the final model prediction confidence across the entire network.

Detecting out-of-distribution samples has been an active research space over the last several years, producing papers that focus on detecting anomalous samples in areas such as image detection [25], [26], text classification [27] and bacteria identification [28].

The ability to detect distribution shifts (away from the training distribution) in sensor data samples at run-time is the key insight of our solution. This detection is hugely beneficial towards efficient AL for indoor localisation that avoids burdening the user excessively.

IV. Distribution Shift Detection With Cross Model Confidence

Here we introduce our proposed solution for detecting data distribution shifts. We determine that a test sample is from a different region to the one used for collecting the training data if there is a data distribution shift. The data distribution shift assessment is made based on the confidence of the prediction.

A. *Confidence Assessment of Model Predictions*

Figure 1 presents the adaptation we make to the neural network model. The prediction model takes a Wi-Fi fingerprint as input and predicts on the final layer the grid cell of the estimated location. At each layer in the network, we add a parallel SoftMax layer. This is trained in a similar manner across the network, using the target class of the training sample to adjust each SoftMax layer. The intuition behind this concept is driven by early exit branches in the GoogleNet neural network. In GoogleNet, if the activation is strong enough on one of these early SoftMax layers along the network, the processing terminates early with that exit its predicted class, before reaching the final layer of the network. In our case, we take the softmax estimations at each level and compute the confidence of the entire model based on their probabilities.

We formulate the model confidence for a given class k and propagating the input x as:

$$confidence_k(x) = \prod_i^n lobit_k^{(i)}(x^{(i)}) \tag{4}$$

where $lobit_k^{(i)}$ is the activation intensity of the SoftMax layer added to layer i for the output associated to class k, and $x^{(i)}$ is the input to layer i as propagated through the network.

This confidence level over the input sample indicates if there is a data distribution shift, if the confidence is lower than a predetermined threshold, or if the sample belongs from the same distribution as the training data if the confidence is higher than the threshold. We determine this threshold empirically based on observations during training.

B. Data Augmentation

We boost the confidence of the classifier for samples in the training set by extending the training set with samples that are very similar to those that are underrepresented. We achieve this by adding very small amounts of noise to the raw sample. In the case of Wi-Fi fingerprints, this is reflected in adding noise chosen from the normal distribution of signal strength deviations observed at a location over many samples in the training set.

Data augmentation is also valuable to reduce the skewness of the data towards more samples for a specific region. By increasing the amount of samples from a region that is underrepresented in the training set, we boost the confidence observed for that class (region) at run-time.

C. Finding the Confidence Threshold

To identify the samples from the unexplored region, a threshold of confidence is imposed on the classifier predictions. After training, the classifier has a weaker confidence score for its predictions on samples from other regions. A deep neural network classifier is trained on the training set (including augmented data).

By analysing the confidence scores from the classifier on the training regions and on the separated region, a threshold of confidence becomes apparent that can differentiate between known training data and the shifted samples.

We then use this threshold of confidence to plot a confidence threshold map, whereby all the samples are plotted by their coordinates, and each sample can be highlighted according to whether it has been singled out as a sample of low confidence, and has been identified as a data shifted sample.

V. Evaluation

A. Dataset Preparation

We use the UJIIndoorLoc dataset [29] for training a position estimation system. This is a multi-building multi-floor indoor localisation dataset to test an Indoor Positioning System that utilises Wi-Fi fingerprints. The UJIIndoorLoc dataset covers 3 buildings of Universitat Jaume I, each having at least 4 floors. Data was gathered using more than 20 different users and 25 Android devices. We perform our evaluation on the samples collected from the ground floor of the leftmost building.

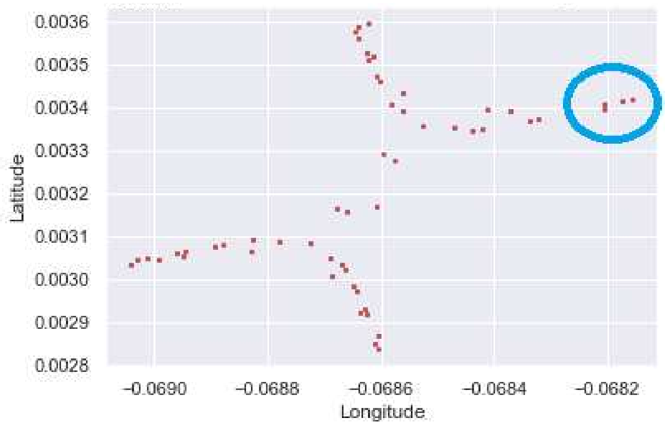

Fig. 2: The leftmost building. The blue circle surrounds the data samples used as the shifted, separate and unexplored region.

B. Splitting the Data into Training and Leave-out Regions

Next, the data samples are split into the main training region and the separate region that is used as the shifted validation data that we use to gauge the performance of the classifier once it is implemented. In Figure 2, the blue circle has been superimposed on the graph to indicate the samples we leave out of the training. For us, this region represents the data distribution shift.

C. Creating the Cell Grid and the Choosing the Classifier

The grid cells are created as 1×1 metre cell size. This splits the geographical area covered by the dataset into approximately 300×200 cells.

A deep neural network classifier is trained on the training data and then tested on the shifted validation data, so that a confidence threshold between the two datasets can be observed. The training hyperparameters include the number of epochs (training iterations) to 500, the mini-batch size is 512, and having just two fully-connected neural network layers.

D. Confidence Score Assessment

After training, the classifier is used to predict the location for samples from the left-out region. For each sample the confidence score is calculated as the likelihood of prediction believes to the sample right grid cell. We observe that on the training data we get high confidence scores. Often the samples are above 80% confidence, and very often above at least 60%. There are some outliers present that score relatively low however these are in the minority and any further training risks overfitting the model.

In Figure 3, the orange lines represent the 'estimated location', i.e. the grid cell classification that the classifier has the

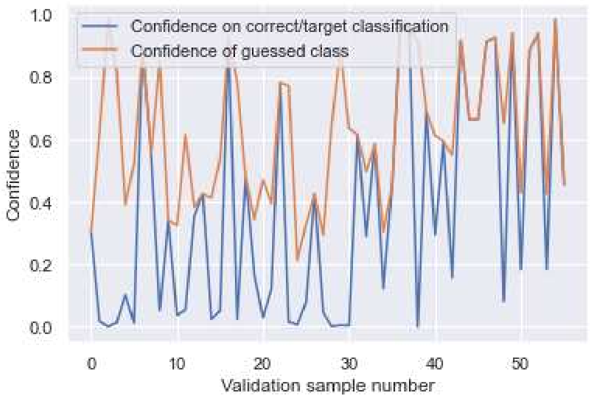

Fig. 3: Confidence scores of classifier on separated and unexplored validation data.

Fig. 4: Distribution of training and validation confidences.

most confidence in for that sample. The blue line represents the confidence of the 'correct location'. The resulting confidence levels show that the classifier usually struggles to predict the correct sample with a high degree of confidence. The blue line shows the confidence scores of the correct classifications, and they usually fall around 0.4 to 0.5. In most cases the confidence levels are below 0.6, and sometimes the confidence scores are very low.

E. Determining the Threshold for Distribution Shift

To detect data distribution shifts, we use a threshold on the classifier confidence. If there is a clear threshold at which point the majority of known samples (training region) are above this confidence level, and the samples from the left-out region are below, this is an effective threshold. Figure 4 presents the box plots to visualise the distribution of the confidences for samples from the training region and samples from the validation (unexplored) region. We see that 0.6 (60%) is an appropriate threshold to use for separating the two regions. This point lies between the lower quartile of the training data and the upper quartile of the validation data, hence most samples are correctly identified as either 'known' locations, or as the anomalous shifted samples we are seeking to detect.

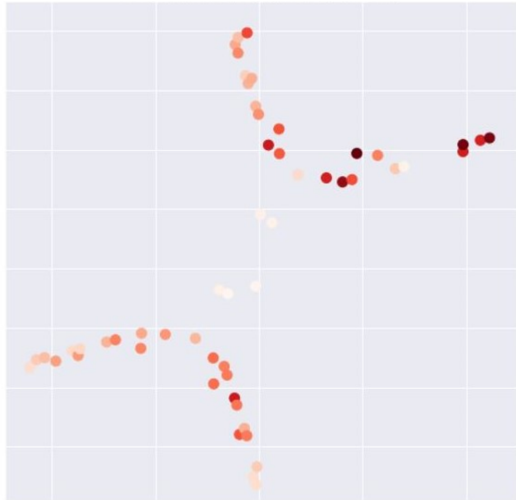

Fig. 5: Classifier confidence on the samples. Darker red indicates *lower* confidence - these are the samples assumed to be shifted.

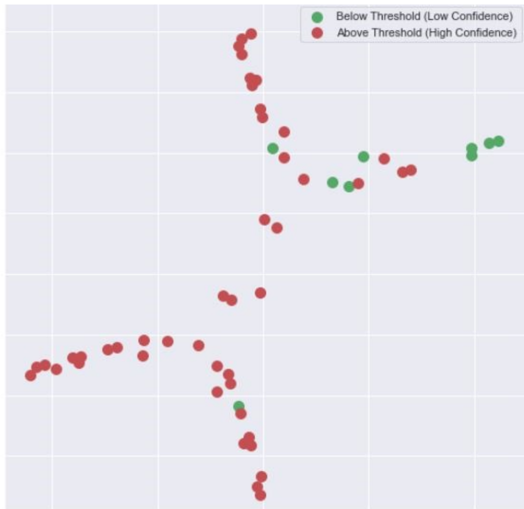

Fig. 6: Separation of samples into what the classifier considers shifted and unshifted data samples.

F. Distribution Shifts for Active Learning

We make use of the same coordinates as before in Figure 2 so we can see how well the classifier performs at identifying the shifted samples. Figure 5 plots the original samples in their locations, shaded to represent the classifier confidence. Figure 5 accurately highlights the data shifted region, the separated samples that are circled in blue in Figure 2. Figure 6 shows the samples once again, but this time the data points are coloured red or green to depict whether the current system considers them a data shifted sample or not.

To reduce the number of false positives, we increase the number of Wi-Fi samples available for the locations of the training set that are wrongly classified. With data augmentation, a small amount of noise is added to the received

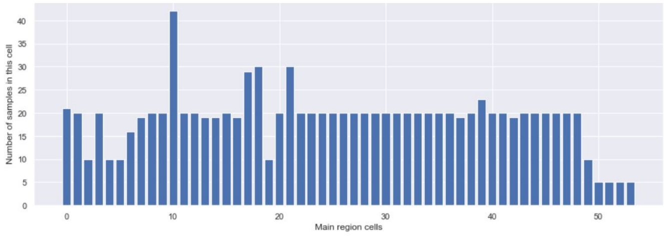

Fig. 7: Bar graph showing the number of Wi-Fi samples available for each grid cell in the extended training set (including data augmentation).

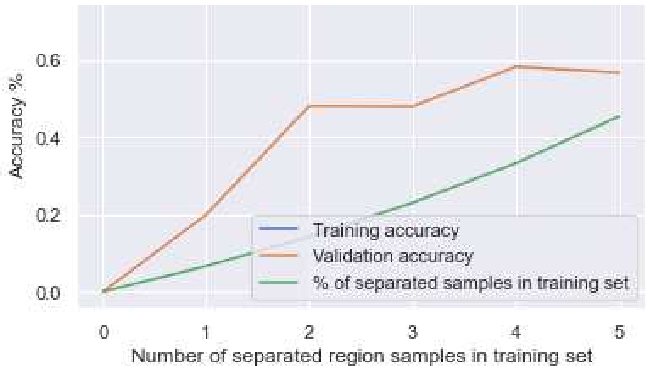

Fig. 8: Accuracy of predictions on training and validation (shifted) data.

signal strengths for the samples collected at those known locations. Figure 7 presents the new distribution in the number of samples for each cell in the training set.

G. Improving Active Learning with Data Distribution Shift Detection

The confidence threshold is fixed at 0.6. We explore how adding samples from the unexplored region influences the quality of the model. Figure 8 presents the effect on the validation set of adding a number of training samples from the unexplored region. This simulates the manual input of users as prompted when inside the unexplored region. This is determined using the data distribution shift technique discussed above.

The orange line shows the increasing accuracy of the classifier as the first few samples from the separated region are added to the training dataset. The increase in accuracy saturates fast, making additional samples irrelevant. This is due to the training generating enough augmented data samples to resemble the real conditions better.

H. Discussion

We showed that our data distribution shift detection method is successful in identifying those samples from isolated regions that represent our data shift. This research shows how to prioritise the Wi-Fi samples that have higher value for an AL algorithm. By using just these relevant samples, we can efficiently update and boost the accuracy of a Wi-Fi fingerprint map.

We can correctly predict all the unseen samples from the separated region as being shifted samples, and these samples are fed into an AL mechanism to update indoor localisation radio maps with user input.

New avenues of research could be explored from the results so far. Whilst we have manually identified and selected a confidence threshold to differentiate between main region and separate region, further investigations could be carried out to determine whether this threshold can be automatically determined to give us the most optimal boundary that still recognises the shifted samples, but minimises the amount of false positives that are created by the classifier.

VI. Conclusions

We showed how Active Learning (AL) can benefit from having an efficient data distribution shift detector to identify the samples that are best suited for user labelling. Our data distribution shift method relies on the confidence score produced by parallel SoftMax layers added to each layer of the location estimation neural network. A threshold imposed on this confidence score indicates which samples are in the comfort zone of the classifier (region experienced in the training set) and which samples are outside of its comfort (region not available in the training set). Data augmentation allows us to boost the confidence of the classifier for those regions that have very few samples in the training set. We show that this can also reduce the amount of samples required to be annotated by users in the AL process.

Our work will be beneficial for updating and maintaining indoor localization system that require up-to-date radio maps and indoor landmark maps. Through this reliable AL solution, the burden imposed on users to provide ground truth inputs is substantially reduced.

Acknowledgement

This document is an overview of UK MOD (part) sponsored research and is released for informational purposes only. The contents of this document should not be interpreted as representing the views of the UK MOD, nor should it be assumed that they reflect any current or future UK MOD policy. The information contained in this document cannot supersede any statutory or contractual requirements or liabilities and is offered without prejudice or commitment.

References

[1] Valentin Radu and Mahesh K Marina. Himloc: Indoor smartphone localization via activity aware pedestrian dead reckoning with selective crowdsourced wifi fingerprinting. In *International conference on indoor positioning and indoor navigation*, pages 1–10. IEEE, 2013.

[2] He Wang, Souvik Sen, Ahmed Elgohary, Moustafa Farid, Moustafa Youssef, and Romit Roy Choudhury. No need to war-drive: Unsupervised indoor localization. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 197–210, 2012.

[3] R. Hansen, R. Wind, C. S. Jensen, and B. Thomsen. Algorithmic strategies for adapting to environmental changes in 802.11 location fingerprinting. In *2010 International Conference on Indoor Positioning and Indoor Navigation*, pages 1–10, 2010.

[4] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. Spotfi: Decimeter level localization using wifi. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pages 269–282, 2015.

[5] Ion Emilian Radoi, Dumitru Cirimpei, and Valentin Radu. Localization systems repository: A platform for open-source localization systems and datasets. In *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8. IEEE, 2019.

[6] Wei Zhang, Kan Liu, Weidong Zhang, Youmei Zhang, and Jason Gu. Deep neural networks for wireless localization in indoor and outdoor environments. *Neurocomputing*, 194:279–287, 2016.

[7] Anca Morar, Alin Moldoveanu, Irina Mocanu, Florica Moldoveanu, Ion Emilian Radoi, Victor Asavei, Alexandru Gradinaru, and Alex Butean. A comprehensive survey of indoor localization methods based on computer vision. *Sensors*, 20(9):2641, 2020.

[8] Adrian Cosma, Ion Emilian Radoi, and Valentin Radu. Camloc: Pedestrian location estimation through body pose estimation on smart cameras. In *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8. IEEE, 2019.

[9] Valentin Radu and Maximilian Henne. Vision2sensor: Knowledge transfer across sensing modalities for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–21, 2019.

[10] Le T Nguyen and Joy Zhang. Wi-fi fingerprinting through active learning using smartphones. In *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, pages 969–976, 2013.

[11] Dilek Hakkani-Tür, Giuseppe Riccardi, and Allen Gorin. Active learning for automatic speech recognition. In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages IV–3904. IEEE, 2002.

[12] Aidan Finn and Nicolas Kushmerick. Active learning selection strategies for information extraction. In *Proceedings of the International Workshop on Adaptive Text Extraction and Mining (ATEM-03)*, pages 18–25, 2003.

[13] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, 2016.

[14] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[15] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Dilan Gorur, Razvan Pascanu, and Hassan Ghasemzadeh. Linear mode connectivity in multitask and continual learning. *arXiv preprint arXiv:2010.04495*, 2020.

[16] Lorena Qendro, Jagmohan Chauhan, Alberto Gil CP Ramos, and Cecilia Mascolo. The benefit of the doubt: Uncertainty aware sensing for edge computing platforms. *arXiv preprint arXiv:2102.05956*, 2021.

[17] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *arXiv preprint arXiv:1612.01474*, 2016.

[18] Burak Turhan. On the dataset shift problem in software engineering prediction models. *Empirical Software Engineering*, 17(1-2):62–74, 2012.

[19] Poojan Oza, Hien V Nguyen, and Vishal M Patel. Multiple class novelty detection under data distribution shift. In *European Conference on Computer Vision*, pages 432–449. Springer, 2020.

[20] Stephan Rabanser, Stephan Günnemann, and Zachary C Lipton. Failing loudly: An empirical study of methods for detecting dataset shift. *arXiv preprint arXiv:1810.11953*, 2018.

[21] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton van den Hengel. Deep learning for anomaly detection: A review. *arXiv preprint arXiv:2007.02500*, 2020.

[22] Saikiran Bulusu, Bhavya Kailkhura, Bo Li, Pramod K Varshney, and Dawn Song. Anomalous example detection in deep learning: A survey. *IEEE Access*, 8:132330–132347, 2020.

[23] Alireza Shafaei, Mark Schmidt, and James J Little. A less biased evaluation of out-of-distribution sample detectors. *arXiv preprint arXiv:1809.04729*, 2018.

[24] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[25] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017.

[26] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

[27] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

[28] Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark A DePristo, Joshua V Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *arXiv preprint arXiv:1906.02845*, 2019.

[29] Joaquín Torres-Sospedra, Raúl Montoliu, Adolfo Martínez-Usó, Tomar J. Arnau, Joan P. Avariento, Mauri Benedito-Bordonau, and Joaquín Huerta. Ujiindoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems. 2014.