Triplet-based Domain Adaptation (Triple-DARE) for Lab-to-field Human Context Recognition

Abdulaziz Alajaji, Walter Gerych, Kavin Chandrasekaran, Luke Buquicchio,
Hamid Mansoor, Emmanuel Agu, Elke Rundensteiner
Worcester Polytechnic Institute, United States
{asalajaji, wgerych, kchandrasekaran, ljbuquicchio, hmansoor, emmanuel,
rundenst}@wpi.edu

***Abstract*—Human Context Recognition (HCR) from smartphone sensor data is an essential task in Context-Aware (CA) systems including those targeting healthcare and security. Two types of smartphone HCR studies (and datasets) have become popular for training HCR models: a) scripted and b) Unscripted/In-the-wild. Supervised machine learning HCR models can achieve good performance on scripted datasets due to their high quality labels but such models generalize poorly to in-the-wild datasets which are more representative of real-world scenarios. In-the-wild datasets are often imbalanced, have missing or wrong labels, with a diversity of phone placements and smartphone models. Lab-to-field approaches try to train HCR models to learn a robust data representation from a high-fidelity, scripted dataset that is used to improve performance on noisy in-the-wild datasets that have similar labels without having to incur the high expense of gathering high-quality labeled dataset. In this paper, leveraging coincident datasets with the same HCR labels collected in separate scripted and unscripted studies, we propose Triplet-based Domain Adaptation for context REcognition (*Triple-DARE*), a novel lab-to-field neural networks method with three key components: 1) a domain alignment loss to learn domain-invariant embeddings, 2) a classification loss to maintain task-discriminative features, 3) a joint fusion triplet loss designed to increase intra-class compactness and inter-class separation in the embedding space of multi-labeled datasets. In rigorous evaluation, *Triple-DARE* improved on the F1-score and classification accuracy of state-of-the-art HCR baselines by 6.3% and 4.5%, respectively, and on HCR models with no adaptation by 44.6% and 10.7%, respectively.**

***Index Terms*—Human context recognition, domain adaptation, ubiquitous computing.**

I. Introduction

Context-Aware (CA) systems that can adapt their behavior based on the user's current context (situation) have significant potential in numerous domains, including healthcare, smart homes, and security systems [1]. Human Context Recognition (HCR), the task of detecting a user's current situation, is essential for CA systems. While several definitions exist in the literature, in this work, we define Human Context as an ⟨Activity, Prioception⟩ tuple, which consists of the user's current activity (e.g., walking, running) and the phone placement (e.g., in a pocket, hand, or bag). We focus on CA and HCR on smartphones that are now ubiquitously owned and are equipped with a rich collection of sensors such as accelerometers, gyroscopes, and position sensors. Two study designs are common in human subjects studies to gather HCR datasets for supervised machine learning 1) *scripted* [2] or 2) *in-the-wild* [3]. In scripted studies, participants perform tasks in a pre-planned order under the supervision of a human proctor while a smartphone app continuously records smartphone sensor readings. Afterward, the human proctors annotate users' sensor data with labels of the contexts they visited. In contrast, *in-the-wild* studies involve collecting data for several days in the real world as subjects live their lives. A smartphone app continuously gathers sensor data and periodically prompts the smartphone owner to report their current context, which is then used to annotate their sensor data.

Issues with In-the-wild datasets that reduce HCR performance: Supervised machine learning classification HCR models typically achieve high accuracy on scripted datasets due to their high-fidelity sensor data and high-quality context labels. For instance, *DeepContext*, a state-of-the-art deep learning HCR model, achieved 91.2% accuracy on a scripted dataset[4]. However, scripted datasets are not realistic as the contexts visited and visit patterns are not representative of real life. It is crucial that HCR models are accurate on in-the-wild datasets, which are more representative of real-world deployment scenarios. However, HCR models achieve lower performance when trained directly on more realistic, in-the-wild datasets. For instance, Vaizman achieved 71.7% accuracy using a Multi-Layer Perceptron (MLP) HCR model trained directly on an in-the-wild dataset. This represents a 19.5% drop in accuracy of state-of-the-art HCR models on scripted vs. in-the-wild datasets. This discrepancy in performance is caused by in-the-wild dataset issues including:

1) Diversity of Phone Placements: or positions in which smartphones are placed (prioceptions). Sensor signals have different signatures for the same activity when the phone is carried in different prioceptions [5]. In fact, prioception is one of the most significant sources of variability in smartphone context sensor data [3], as illustrated in Figure (2). Smartphone users may choose to carry their smartphones in a bag, in their hand, or in their coat pocket while performing a given activity (e.g., walking).

2) Weak, noisy, and missing context labels: as users stop providing labels when their lives get busy, or they may erroneously provide wrong labels [6], which presents a challenge for supervised machine learning algorithms [7].

3) Diversity of smartphone models: Unlike scripted HCR studies where subjects use a single study phone model provided by the proctor, subjects in in-the-wild HCR studies typically use their own phones. The sensor values recorded

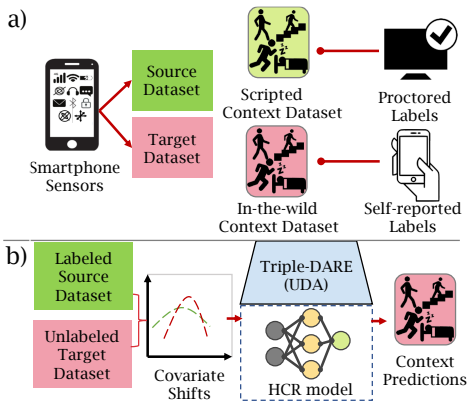

Fig. 1: a) The nature of the two smartphone context data we use in this work. b) A high-level overview for *Triple-DARE*'s problem and approach.

for a given context by different smartphone models can differ by as much as 30% [8], presenting an additional challenge for machine learning classifiers.

Lab-to-field methods: have recently emerged as viable solutions to achieve good HCR performance on in-the-wild datasets that have noisy, low-quality labels [9]. Lab-to-field approaches try to train highly accurate machine learning models on scripted datasets, which are adapted for *in-the-wild* datasets with the hope of maintaining good performance. However, in general, the performance of HCR models trained naively on scripted datasets often drops when tested on in-the-wild datasets *(Going from Lab-to-field)*. This performance drop is because in addition to the in-the-wild dataset issues listed above, the contexts visited by subjects in the scripted study, as well as their context visit order and visit duration differ significantly from in-the-wild scenarios. Consequently, there are significant differences between the distribution of features extracted from scripted vs. in-the-wild datasets, also known as the *covariate shift problem* [9]–[11].

Domain Adaptation (DA) is a transductive transfer learning method and one of the main solutions used to adapt neural networks to mitigate the covariate shift problem. DA has been employed in various related domains, including object detection in computer vision and the problems caused by the variability of wearable sensor placement in ubiquitous computing[5], [11]. Unsupervised DA (UDA) tries to learn a deep learning model using a combination of a labeled source (e.g. scripted dataset) and unlabeled target (e.g. in-the-wild) samples with different distributions to achieve accurate predictions on previously unseen, unlabeled (e.g. in-the-wild) samples [5], [12]. Figure (1) provides a high-level overview of the problem, challenges, and our approach.

Challenges. Two key challenges must be addressed for using UDA for Lab-to-Field generalization of smartphone context recognition. First, the previously described data issues with in-the-wild datasets (the diversity of placements, weak and noisy labels, and diverse smartphone types) must be overcome. Secondly, it is challenging to develop a robust method for transferring knowledge from a scripted dataset to

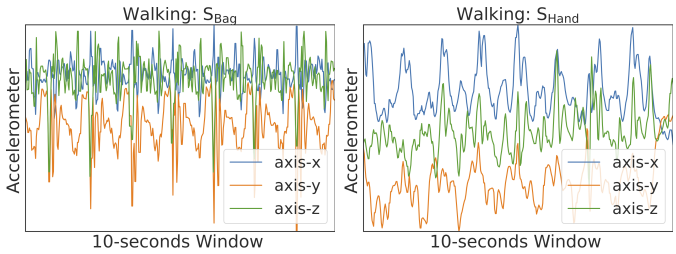

Fig. 2: The influence of diverse phone placements on sensor data is observable in the triaxial accelerometer signal for the same walking activity but with different prioceptions.

a more realistic but considerably noisier in-the-wild dataset with sparse labels.

Our approach. We are motivated by the recent empirical success of triplet loss in face identification [13], [14], where variations of the same person's face images are mapped closely in the learned embedding space. We believe sensor data can benefit from the same approach where there is often variation in sensor signatures corresponding to the same context. Our belief is also consistent with Khaertdinov et al.'s findings where triplet loss was applied recently to mitigate the effects of subject heterogeneity and improve model generalizability [15].

We propose *Triple-DARE*, a deep Lab-to-field UDA method that is able to leverage the tremendous amounts of unlabeled in-the-wild smartphone HCR data, decreasing the need for human-annotated labels. To facilitate our DA approach, we utilized coincident scripted and in-the-wild HCR datasets in which similar context labels were gathered in both studies [1]. These coincident datasets and similar context labels ensure that there exists a feature representation of contexts that is common between the scripted and in-the-wild datasets, a key requirement for the DA approach. We demonstrate our method's applicability to HCR models deployed in realistic environments by using context labels gathered in a scripted study only during model development and using DA to mitigate the influence of potentially noisy labels and retain HCR performance on an in-the-wild dataset. *Triple-DARE* outperforms state-of-the-art baselines with 3.79% and 1.89% increases in F1-score and classification accuracy, respectively, and also achieves improvements of 39% and 14.7% in F1-score and classification accuracy, respectively, HCR models without *Triple-DARE*.

Contributions. The main contributions of this paper are:

1) We propose *Triple-DARE*, a novel UDA deep-learning framework, which uses a scripted dataset is to improve the HCR accuracy of predicting in-the-wild contexts. *Triple-DARE* utilizes a domain alignment loss to learn domain-invariant features, a classification loss to maintain task-discriminative features, and a joint fusion triplet loss to increase intra-class compactness and inter-class separation.
2) We rigorously evaluated *Triple-DARE*, comparing it to multiple state-of-the-art unsupervised domain methods,

including DAN[16], CORAL[17], and HDCNN[18], and bench-marking improvements in HCR performance on target domains in several use cases.

3) We demonstrate that *Triple-DARE* mitigates in-the-wild dataset challenges when compared to state-of-the-art DA methods, achieving high prediction performance on the target (in-the-wild) domain without the need for large amounts of source labeled samples.

The remainder of this paper is organized as follows. Section II lists related work. Section III introduces our proposed approach. Section IV demonstrates our evaluation and results. Finally, Section V concludes the paper.

II. Related work

Lab-to-field generalization. Lab-to-field methods previously proposed to handle covariate shifts include importance re-weighting[9], [19] and Positive Unlabeled (PU) classifiers[1]. There is very little work on lab-to-field generalization for HCR. However, a related work that dealt with this problem on wearable electrocardiogram (ECG) data [9], used importance re-weighting to adapt a linear logistic regression model. However, these methods have achieved lower performance when applied to deep neural networks[20].

DA for wearable sensor data. Several DA methods have been proposed to adapt a trained model for use on another related dataset in ubiquitous computing [5], [18], [21]. DA has previously been used to address the issue of variability in the placement of wearable sensors, learning domain-invariant accelerometer [5], [18] and gyroscope[5] features from sensor data by minimizing a discrepancy distance in the Convolutional Neural Network (CNN) embedding. HDCNN[18] investigated adapting a pre-trained model on smartphone data to work on unlabeled smartwatch data. A discrepancy-based approach using Kullback-Leibler (KL) divergence was employed to adapt the model that was pre-trained on smartphone data, for unlabeled smartwatch data. Prior work focuses only on reducing the global distribution discrepancy while learning common feature representations across domains [5], [18]. Our work builds on prior work by utilizing a joint fusion triplet loss to improve intra-class compactness and inter-class separability [12], [13].

III. Proposed *Triple-DARE* Methodology

A. Problem Formulation

In this work, we utilize data from two coincident datasets in which similar context labels were gathered: 1) a scripted dataset (source) with high-quality labels and 2) an in-the-wild dataset (target) with similar context ($\langle$Activity, Phone Prioception$\rangle$) labels, shown in Table II. Initially, the HCR model is trained using the labeled source dataset. Afterwards, the trained HCR model is utilized to recognize unlabeled contexts in the target dataset. With regards to UDA, there are labeled samples for the source domain and unlabeled samples for the target domain, which have different data distributions. Our goal is to learn a classifier from a combination of labeled source and unlabeled target data, which generalizes well on the target domain. Formally, we have labeled samples $D_s = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^{n_s}$ and unlabeled samples $D_t = \{(\mathbf{x}_i^t)\}_{i=1}^{n_t}$ where n_s and n_t represent the number of samples in the source and target domains, respectively. Both the source and the target domain share the same feature space $\mathcal{X}_s = \mathcal{X}_t$ and label space $\mathcal{Y}_s = \mathcal{Y}_t$, but differ in the marginal distribution ($P_s(x_s) \neq P_t(x_t)$). The conditional distributions are presumed to be equal $P_s(y_t|x_s) = P_t(y_t|x_t)$. We denote $\boldsymbol{x}$ as a feature vector and $\boldsymbol{y}$ as a human context represented by a multi-label output vector, where each label produced is a binary output (E.g walking vs not walking). The source and target tasks are presumed to be the same.

B. Overview

As illustrated in Figure (3), *Triple-DARE* has two types of feature sources extracted from both the source scripted and target in-the-wild datasets : 1) Time and frequency based handcrafted features handled by a feed-forward network and 2) raw three-axial sensors fed into an attention-based CNN that extracts salient features from raw sensor data using a soft attention mechanism. *Triple-DARE* has three major learning components: 1) A domain alignment loss $\mathcal{L}_{\mathrm{d}}$ to extract embeddings that are invariant across domains. 2) a classification loss $\mathcal{L}_{\mathrm{cls}}$ to maintain task-discriminative features, 3) a joint fusion triplet loss $\mathcal{L}_{\mathrm{tri}}$ to increase intra-class compactness and inter-class separation in the learned embedding space by learning similar contexts represented by variations of sensor inputs. *Triple-DARE*'s final output is used for multi-labeled context predictions. For example, based on our <Activity, Phone placement> definition of context, example contexts include $\langle$"Sitting","In Bathroom" with "Phone In Hand"$\rangle$. In order to perform our context predictions by learning discriminative and domain-invariant embeddings, our final objective is to minimize the cost function $C(\cdot)$

$$C(\theta) = \lambda_1 \mathcal{L}_{\mathrm{cls}}^{\theta} + \lambda_2 \mathcal{L}_{\mathrm{d}}^{\theta} + \lambda_3 \mathcal{L}_{\mathrm{tri}}^{\theta}, \tag{1}$$

where θ are model parameters, λ_1, λ_2, and λ_3 are balancing coefficients. Each of these types of losses are described in more detail in subsequent subsections.

C. Feature Generation

For a given smartphone context dataset, two views are created from raw sensor input data. View 1 is a vector obtained by extracting handcrafted features from all available sensors. View 2 consists of raw tri-axial sensor data. Different feature encoders were used for each type of input view: 1) Multi-Layer Perceptron (MLP) encoder for handcrafted features, and 2) An attention-based CNN encoder for raw sensor data. Finally, a joint fusion encoding is obtained by concatenating the two feature encodings generated. In work on our prior DeepContext HCR model, we found this two-view feature generation approach to be effective [4].

We use data from 5 sensors: accelerometer, gyroscope, GPS, magnetometer, and phone state (e.g., is phone screen locked/unlocked?). We compute statistical, time- and frequency-based features for each sensor type at a sliding

Feature	Formulation
Tri-axial sensors Features	
Arithmetic mean	$\bar{s} = \frac{1}{N}\sum_{i=1}^{N} s_i$
Standard deviation	$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(s_i - \bar{s})^2}$
Frequency signal Skewness	$\mathbb{E}\left[\frac{(s-\bar{s})^3}{\sigma}\right]$
Frequency signal Kurtosis	$\mathbb{E}\left[(s-\bar{s})^4\right] / \mathbb{E}\left[(s-\bar{s})^2\right]^2$
Signal magnitude area	$\frac{1}{3}\sum_{i=1}^{3}\sum_{j=1}^{N}\lvert s_{i,j}\rvert$
Pearson Correlation coefficient	$C_{1,2}/\sqrt{C_{1,1}C_{2,2}}, C = \mathrm{cov}(s_1, s_2)$
Spectral energy of a frequency band [a, b]	$\frac{1}{a-b+1}\sum_{i=a}^{b} s_i^2$
s: signal vector, N: signal vector length Q: quartile, cov: covariance	
GPS Features	
Significant changes from the previous location state	
Estimated speed	
Changes in latitude and longitude	
Phone State Features	
Is phone screen unlocked?	Is battery charging?
Is ringer mode set to silent?	Is phone connected to WIFI?

TABLE I: A sample list of handcrafted features extracted from accelerometer, gyroscope, and magnetometer sensor data, adopted from [3], [22].

window level. A 10-second window was selected based on findings of prior work [4]). Z-score normalization $z_i = \frac{x_i - \bar{x}}{s}$ was then applied. These 188 handcrafted features [3] were used to construct a vector that was fed into a feed-forward network.

The CNN auto-learns a representation from raw sensor data from three axial sensors (accelerometer, gyroscope, and magnetometer). Adapted from our previous work, DeepContext [4], the CNN we leveraged has a soft attention mechanism that learns salient features, while giving higher weights to regions of the raw sensor data that are more predictive of the user's context. As a result, the model is able to highlight discriminative CNN features or different contexts.

D. Domain Alignment Loss

The goal of the domain alignment loss is to map the source and target feature encodings into a standard feature distribution space and learn common feature representations across domains. We utilized Multi Kernel Maximum Discrepancy Mean (MK-MMD), an extension for Maximum Discrepancy Mean (MMD) [23] as our domain alignment loss. MMD is a non-parametric distance measure that may be used to assess the discrepancy between marginal distributions [16].

The formulation of MK-MMD is defined as:

$$\mathrm{q}(\mathcal{X}^s, \mathcal{X}^t) = \left\| E_{\mathcal{X}^s}\left[\phi\left(x^s\right)\right] - E_{\mathcal{X}^t}\left[\phi\left(x^t\right)\right] \right\|_H, \quad (2)$$

where $\|.\|_{H_k}$ is the RKHS norm and $\phi(\cdot)$ is a feature map defined as a combination of multiple positive kernels. The

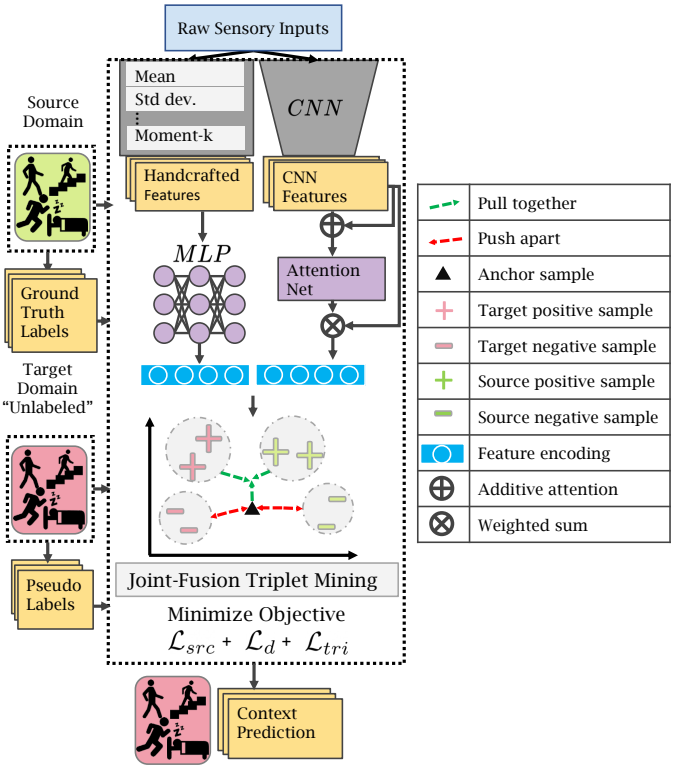

Fig. 3: *Triple-DARE* Framework.

domain alignment loss can be obtained by:

$$\mathcal{L}_{\mathrm{d}}^{\theta} = \sum_{l \in N^l} \mathrm{q}^2\left(\mathcal{X}_l^s, \mathcal{X}_l^t\right). \quad (3)$$

MK-MMD is calculated per layer of the network to quantify the distance between the data representation for the source and target domains. N^l is the number of layers, and $(\mathcal{X}_s^l, \mathcal{X}_t^l)$ are the distributions of the source and target domains, extracted from the lth layer in the network.

E. Classification Loss

The classification loss aims to leverage source domain labels in discovering features that are discriminative for context predictions. Optimizing our model for classifying contexts on the source domain guides the overall learning process. Since the labels of D_s are available, the classification loss is defined as:

$$\mathcal{L}_{\mathrm{cls}}^{\theta} = \frac{1}{N_s}\sum_{i=1}^{N_s} \ell_{\Psi}(f_{\phi}(x_i^s), y_i^s), \quad (4)$$

where $f_{\phi}(\cdot)$ is a classifier, N_s is the number of labeled training samples and ℓ_{Ψ} is a binary cross-entropy function weighted by inverse class frequency to account for class imbalance where infrequently-occurring classes get higher weights than frequently occurring classes.

F. Triplet Loss

The triplet loss is used to pull samples belonging to the same or similar classes together, while pushing away samples belonging to different classes in an embedding space[13], [14]. Given three types of samples: 1) an anchor sample x_a (i.e. a

query sample), 2) a positive sample x_p belonging to the same class as the anchor, and 3) a negative sample x_n belonging to a different class from the anchor. Along with a distance function d, triplet loss is defined as the following:

$$\mathcal{L}_{\text{tri}}^{\theta} = \sum_{i}^{N} \left[d(x_a^i, x_p^i) - d(x_a^i, x_n^i) + \alpha \right]_+ \tag{5}$$

Where α is a parameter for the margin between positive and negative samples, and x is used here to represent an embedding of x for notation simplicity. Pairs of positive samples are pulled together jointly, while pairs of positive and negative samples are pushed away by some margin α.

G. Joint-Fusion Triplet Mining

Triplet mining is the process of constructing triplets (anchor, positive and negative) for triplet loss calculations. Following the practice in [13], we adopt the online triplet mining strategy which does not require a complete pass on the training set beforehand. Since finding triplets across two domains requires the existence of target domain labels, the classifier trained on the source domain is used to construct pseudo labels for target domain samples during training the classifier, which is one of the most common solutions for UDA problems. [12]. We re-assign pseudo labels every few iterations because the classifier accuracy on the target dataset increases steadily during training. Additionally, the domain alignment loss can also improve the classifier's accuracy on the target dataset by lowering the distribution disparity. As a result, the quality of the pseudo label can automatically improve. We create triplets from two mini-batches of samples from the source and target domains after concatenating them into one mini-batch. For constructing triplets suitable to our multi-labeling settings, we need a notion of similarity between multi-labeled vectors. We first define a compatibility score between two contexts y_1, y_2 that are both represented as binary labels, as the dot product between them:

$$c(y_1, y_2) = y_1 \cdot y_2 \tag{6}$$

Because our dataset is extremely imbalanced, we consider all the positive examples in constructing the triplets. We follow a similar strategy to [13] that focuses on triplets contributing the most to the learning process but modified using our compatibility score to select triplets that satisfy this condition:

$$d(x_a, x_p) + \alpha > d(x_a, x_n) \;\; \& \;\; c(y_a, y_p) > c(y_a, y_n) \tag{7}$$

IV. Experiments

We evaluated Triple-DARE and baseline models for performing multiple UDA use cases on coincident scripted and in-the-wild smartphone HCR datasets. The overarching goal was to use *Triple-DARE* to learn a robust representation from the scripted dataset (source), which is then used to improve HCR on the in-the-wild dataset (target).

TABLE II: The percentage of positively labeled contexts.

Contexts	**Scripted % P**	**In-the-wild % P**
Bathroom	3.15%	2.17%
Jogging	2.04%	0.27%
Lying Down	1.10%	16.24%
Running	1.95%	0.37%
Sitting	11.99%	38.71%
Sleeping	2.19%	37.69%
Stairs - Going Down	2.52%	2.00%
Stairs - Going Up	0.89%	1.92%
Standing	1.71%	8.46%
Talking On Phone	1.41%	1.27%
Typing	3.65%	6.45%
Walking	64.00%	13.51%
Phone Prioceptions		
Phone In Hand	Phone In Pocket	Phone In Bag
Datasets Notations		
$S_{Prioception}$	Scripted context dataset	
$W_{Prioception}$	In-the-wild context dataset	
e.g. S_{Bag} refers to scripted contexts, annotated with "Phone In Bag"		

A. Datasets

In-the-wild dataset: 103 participants downloaded a smartphone app that passively collected data from their smartphones as they lived their lives for two weeks. Participants were periodically asked to self-report the labels of the contexts they visited, listed in Table (II). By acquiring data utilizing individuals' smartphones, our in-the-wild dataset reflected a diversity of manufacturer hardware and contexts visited by the user were realistic. *Scripted dataset:* The scripted study was conducted in specific buildings, laboratories, or routes on a college campus. The smartphone app collected data from 100 participants that visited pre-planned contexts. Human proctors oversaw and manually annotated the data during the data collection session, which lasted about an hour per subject.

Data pre-processind and feature extraction: Both the scripted and in-the-wild datasets were pre-processed and featurized in the same way. Contexts were handled as multi-label vectors, with segments created using a 10-second window size. The number of samples was 21,846 and 631,026 for the scripted and in-the-wild datasets, respectively. Table (II) lists the context labels in both datasets. To increase the generalizability of our model to unseen subjects, we adopted subject-wise cross-validation in which all of a subject's data appeared either in the training or test sets but not both. In each UDA experiment, 90% of source domain data was used for training, 10% of source data was reserved for validation, and all data in the target domain was used for testing.

B. Baselines

We compared *Triple-DARE* to state-of-the-art deep-learning based DA models : 1) *CORAL*[17]: A UDA model that utilizes deep-coral discrepancy loss. 2) *DAN*[16]: A model with only our MK-MMD domain alignment loss. 3) *HDCNN*[18]: a state-of-the-art baseline DA method previously applied on smartphone sensor data. *HDCNN* is a DA method with KL divergence loss on the feature vectors obtained across domains. 4) *SOURCE*: A model trained on the source domain without any adaptation to the target domain. Our proposed model, *Triple-DARE*, which uses our joint-fusion triplet loss.

C. Implementation and Experimental Settings

1) Hyper-parameters Grid search was used to tune the hyper-parameters of the MLP and CNN. The learning rate

TABLE III: Overall context prediction

Overall UDA Tasks	Accuracy	F1-micro
Triple-DARE	**0.879**	**0.366**
CORAL	0.806	0.302
DAN	0.673	0.294
HDCNN	0.816	0.3215
Source (no adaptation)	0.433	0.259
Lab-to-field UDA Tasks	**Accuracy**	**F1-micro**
Triple-DARE	**0.845**	**0.188**
CORAL	0.839	0.127
DAN	0.698	0.122
HDCNN	0.768	0.146
Source (no adaptation)	0.552	0.133

was initialized to 1e-1, balancing coefficients were initialized as $\lambda_1 = 1$, $\lambda_2 = 0$, and $\lambda_3 = 0$. The balancing coefficients and the learning rate were increased or decreased following the schedule mentioned in [24], making our model highly confident on source labels and less sensitive to low-quality pseudo labels at the early stages of the training. The batch size was set to 256 and the Adam optimizer was used. The back-bone layers used in our DA method were shared across all experiments: handcrafted-features with a 2-layer MLP, each layer having 16 hidden dimensions and an MLP domain classifier with one layer with 32 hidden dimensions, and CNN with attention blocks for separate and merged sensors layers, followed by an average pooling layer, adopted from [4]. All raw sensor data were input to a 3-layer CNN. Then their outputs are concatenated and forwarded to another 3-layer CNN. Attention blocks are used to focus on salient regions in inputs [4], [25]. Euclidean distance was used for computing pairwise distances in triplet mining and α was set to 0.1. The final context prediction layer has LeakyReLU activation, followed by Sigmoid activation.

2) Evaluation Protocol Due to the class imbalance in our context datasets, in addition to classification accuracy, we used the *F1* metric to evaluate HCR performance in the UDA setting. As the sizes of the source and target domain datasets may not be the same, we iterate through the target domain dataset with random sampling. However, we evaluate our model on all samples in the target domain dataset.

D. Results and Findings

1) Overall Results: The overall performance scores for our *Triple-DARE* compared to baseline models is reported in Table (III). *Triple-DARE* outperforms the baseline methods in the overall UDA tasks and Lab-to-field UDA task by 4.5% increase in F1-score and 6.3% increase in classification accuracy. The result, shown in Figure (4), demonstrate the performance per label aggregated over all the UDA tasks, showing that our approach outperforms state-of-the-art methods across several context labels. In general, the advantage of using UDA methods can be observed over classifiers that are solely trained on the source domain without leveraging unlabeled data. In particular, UDA methods helped a lot in the Jogging, Running, Going Up and Down Stairs labels where the user is likely to stop providing labels while performing these activities in the wild. *Triple-DARE* improves adaptation using the high fidelity labels acquired in the scripted study.

2) Training under insufficient labels: As presented in Table (IV), we studied the performance of our model when the number of labels from the source domain is varied. Triple-DARE achieves higher prediction scores on the target domain, with small amounts of source labels, outperforming baseline methods in almost all UDA tasks.

3) Intra-class compactness and inter-class separation: To provide a measure of compactness and separation in the learned feature embeddings, we utilized the Silhouettescore Score $= \frac{b_i - a_i}{\max(b_i, a_i)}$, where b_i is the shortest mean distance between a point to all other points in any other cluster, whereas a_i is the mean distance of i and all data points from the same cluster. This score measures both compactness and separation. To calculate the Silhouette scores on the learned feature embeddings, we assign each instance with cluster labels using one of the binary context labels. Then, we averaged the scores over labels. The scores are reported in Figure (5), which shows that *Triple-DARE* achieves better compactness and separation scores in most UDA tasks. CORAL achieves higher scores than DAN in most cases.

V. Conclusion

Several issues reduce the performance of machine learning HCR models on in-the-wild datasets, including the diversity of phone placements and smartphone models and weak, noisy, or missing labels. Lab-to-field methods try to improve the performance of HCR models by first training them on similar scripted datasets, then adapting them for use in predicting context labels in in-the-wild datasets. We designed DA strategies that are susceptible to covariate shifts between the scripted and in-the-wild datasets, improving lab-to-field generalization. This paper proposed Triple-DARE, a UDA deep-learning model for HCR on smartphones, comprised of three parts: 1) domain alignment loss using MK-MMD 2) a classification loss and, 3) joint-fusion triplet loss designed for multi-labeled datasets. Triple-DARE learns domain-invariant features common to both datasets, reducing the influence of highly noisy in-the-wild data by using its attention mechanism to focus on salient regions in sensor inputs, achieving a high F1-score for various UDA tasks on our scripted and in-the-wild context datasets. Using its domain alignment loss, Triple-DARE is able to map the source and target feature encoding into a standard feature distribution space with better performance than state-of-the-art baseline methods. Furthermore, the triplet loss improves discrimination, increasing intra-class compactness and inter-class separation while leveraging massive amounts of unlabeled data. *Triple-DARE* outperforms other state-of-the-art DA baselines, improving on their F1-score and classification accuracy by 4.6% and 1.89%, respectively, and improving on models with no adaptations by 10.7% and 14.7%, respectively. In future work, we plan to leverage our proposed method in representation learning for smartphone sensor data. One limitation in our model is the assumption that an identical number of sensors are available in both scripted and in-the-wild datasets. We could improve our framework by increasing model robustness against missing sensors during model inference.

TABLE IV: F-1 scores - comparing different methods for various UDA tasks, varying the amounts of source labels used.

		Scripted contexts with cross-prioception UDA tasks							Lab-to-field UDA Tasks			
Training %	Method	$S_{Bag} \rightarrow S_{Hand}$	$S_{Bag} \rightarrow S_{Pocket}$	$S_{Hand} \rightarrow S_{Bag}$	$S_{Hand} \rightarrow S_{Pocket}$	$S_{Pocket} \rightarrow S_{Bag}$	$S_{Pocket} \rightarrow S_{Hand}$	Avg	$S_{Bag} \rightarrow W_{Bag}$	$S_{Hand} \rightarrow W_{Hand}$	$S_{Pocket} \rightarrow W_{Pocket}$	Avg
0.2	*Triple-DARE*	**0.500**	**0.651**	0.213	0.318	**0.652**	**0.467**	**0.467**	**0.101**	0.080	**0.326**	**0.169**
	CORAL	0.357	0.328	0.357	0.428	0.352	0.378	0.367	0.089	**0.087**	0.150	0.109
	DAN	0.341	0.436	0.285	0.265	0.418	0.403	0.358	0.079	0.077	0.165	0.107
	HDCNN	0.341	0.492	**0.472**	**0.470**	0.468	0.380	0.437	0.087	0.084	0.181	0.117
0.4	*Triple-DARE*	**0.557**	**0.617**	0.444	**0.492**	**0.767**	**0.511**	**0.565**	**0.118**	**0.143**	**0.359**	**0.207**
	CORAL	0.380	0.584	**0.455**	0.457	0.633	0.484	0.499	0.092	0.075	0.165	0.111
	DAN	0.452	0.509	0.418	0.451	0.721	0.459	0.502	0.101	0.093	0.244	0.146
	HDCNN	0.424	0.580	0.504	0.558	0.704	0.441	0.535	0.106	0.108	0.266	0.160
0.6	*Triple-DARE*	0.497	0.588	**0.570**	**0.653**	0.744	**0.542**	**0.599**	**0.111**	0.112	**0.341**	**0.188**
	CORAL	**0.577**	**0.688**	0.505	0.505	**0.754**	0.448	0.580	0.110	**0.123**	0.210	0.148
	DAN	0.540	0.634	0.428	0.459	0.653	0.429	0.524	0.100	0.084	0.209	0.127
	HDCNN	0.345	0.561	0.575	0.540	0.645	0.445	0.518	0.094	0.102	0.285	0.160
Average	*Triple-DARE*	**0.518**	**0.619**	0.409	0.488	**0.721**	**0.507**	**0.544**	**0.111**	**0.112**	**0.341**	**0.188**
	CORAL	0.438	0.533	0.439	0.463	0.580	0.437	0.482	0.097	0.093	0.173	0.122
	DAN	0.440	0.526	0.377	0.392	0.597	0.430	0.461	0.096	0.087	0.198	0.127
	HDCNN	0.370	0.544	**0.517**	**0.523**	0.606	0.422	0.497	0.096	0.098	0.244	0.146
-	No Adaptation	0.319	0.469	0.476	0.470	0.260	0.315	0.385	0.108	0.110	0.180	0.133

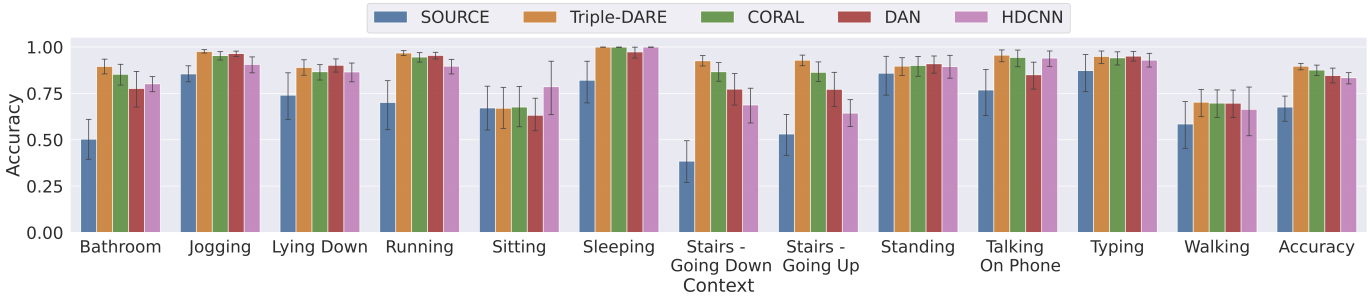

Fig. 4: Target predictions scores per label averaged across different UDA task domains

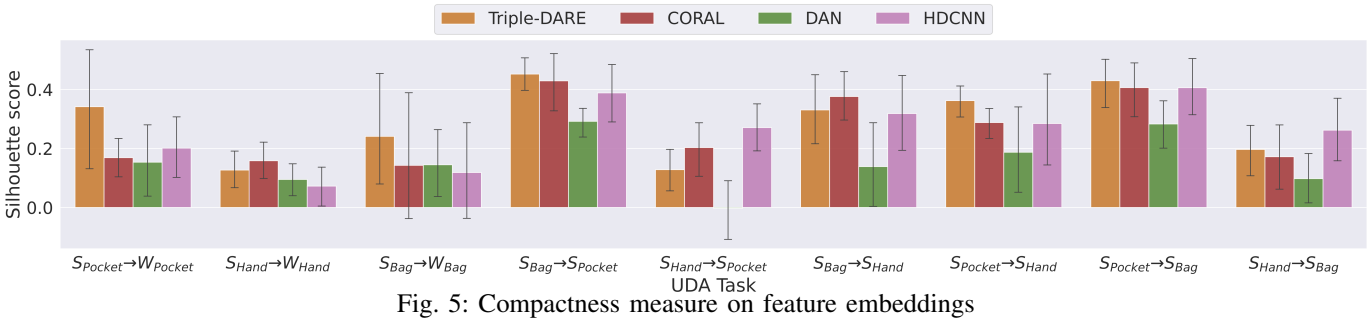

Fig. 5: Compactness measure on feature embeddings

VI. Acknowledgment

This material is based on research sponsored by DARPA under agreement number FA8750-18-2-0077. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

References

[1] A. Alajaji, W. Gerych, L. Buquicchio, K. Chandrasekaran, H. Mansoor, E. Agu, and E. A. Rundensteiner, "Smartphone health biomarkers: Positive unlabeled learning of in-the-wild contexts," *IEEE Pervasive Computing*, vol. 20, pp. 50–61, 2021.

[2] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *Proc. INSS 2010*. IEEE, 2010, pp. 233–240.

[3] Y. Vaizman, K. Ellis, and G. Lanckriet, "Recognizing detailed human context in the wild from smartphones and smartwatches," *IEEE Pervasive Computing*, vol. 16, no. 4, pp. 62–74, 2017.

[4] A. Alajaji, W. Gerych, K. Chandrasekaran, L. Buquicchio, E. Agu, and E. Rundensteiner, "Deepcontext: Parameterized compatibility-based attention cnn for human context recognition," in *2020 IEEE 14th International Conference on Semantic Computing (ICSC)*, 2020.

[5] Y. Chang, A. Mathur, A. Isopoussu, J. Song, and F. Kawsar, "A systematic study of unsupervised domain adaptation for robust human-activity recognition," *ACM IMWUT*, vol. 4, pp. 1–30, 03 2020.

[6] H. Mansoor, W. Gerych, L. Buquicchio, K. Chandrasekaran, E. Agu, and E. Rundensteiner, "Delfi: Mislabelled human context detection using multi-feature similarity linking," in *2019 IEEE VDS*, 2019, pp. 11–19.

[7] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *National Science Review*, vol. 5, no. 1, pp. 44–53, 2018.

[8] A. Stisen, H. Blunck, S. Bhattacharya, T. Prentow, M. Kjærgaard, A. Dey, T. Sonne, and M. Jensen, "Smart devices are different: Assessing and mitigating mobile sensing heterogeneities for activity recognition," in *Proc. Sensys*, 11 2015, pp. 127–140.

[9] A. Natarajan, G. Angarita, E. Gaiser, R. Malison, D. Ganesan, and B. M. Marlin, "Domain adaptation methods for improving lab-to-field generalization of cocaine detection using wearable ecg," in *Proc. Ubicomp*. New York, NY, USA: ACM, 2016, p. 875–885.

[10] "A unifying view on dataset shift in classification," vol. 45, no. 1, pp.

521–530, 2012.
[11] W. M. Kouw, "An introduction to domain adaptation and transfer learning," *ArXiv*, vol. abs/1812.11806, 2018.
[12] W. Deng, L. Zheng, and J. Jiao, "Domain alignment with triplets," 12 2018.
[13] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," *IEEE CVPR*, pp. 815–823, 2015.
[14] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *ArXiv*, vol. abs/1703.07737, 2017.
[15] B. Khaertdinov, E. Ghaleb, and S. Asteriadis, "Deep triplet networks with attention for sensor-based human activity recognition," in *2021 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2021, pp. 1–10.
[16] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," *ArXiv*, vol. abs/1502.02791, 2015.
[17] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *ECCV Workshops*, 2016.
[18] M. A. A. H. Khan, N. Roy, and A. Misra, "Scaling human activity recognition via deep learning-based domain adaptation," in *2018 IEEE PerCom*, 2018, pp. 1–9.
[19] H. Hachiya, M. Sugiyama, and N. Ueda, "Importance-weighted least-squares probabilistic classifier for covariate shift adaptation with application to human activity recognition," *Neurocomputing*, vol. 80, pp. 93–101, 2012, special Issue on Machine Learning for Signal Processing 2010.
[20] J. Byrd and Z. C. Lipton, "What is the effect of importance weighting in deep learning?" in *ICML*, 2019.
[21] "Cross-position activity recognition with stratified transfer learning," *Pervasive and Mobile Computing*, vol. 57, pp. 1–13, 2019.
[22] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-Aware Human Activity Recognition Using Smartphones," *Neurocomputing*, vol. 171, pp. 754–767, Jan. 2016. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0925231215010930
[23] A. Gretton, B. K. Sriperumbudur, D. Sejdinovic, H. Strathmann, S. Balakrishnan, M. Pontil, and K. Fukumizu, "Optimal kernel choice for large-scale two-sample tests," in *NIPS*, 2012.
[24] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, pp. 59:1–59:35, 2016.
[25] S. Jetley, N. A. Lord, N. Lee, and P. H. S. Torr, "Learn To Pay Attention," *arXiv:1804.02391 [cs]*, Apr. 2018, arXiv: 1804.02391. [Online]. Available: http://arxiv.org/abs/1804.02391