Hierarchical Active Learning for Model Personalization in the Presence of Label Scarcity

Annamalai Natarajan
Philips Research, Cambridge, MA
Annamalai.Natarajan@philips.com

Deepak Ganesan
University of Massachusetts Amherst
dganesan@cs.umass.edu

Benjamin M. Marlin
University of Massachusetts Amherst
marlin@cs.umass.edu

***Abstract*—In mobile health (mHealth) and human activity recognition (HAR), collecting labeled data often comes at a significantly higher cost or level of user burden than collecting unlabeled data. This motivates the idea of attempting to optimize the collection of labeled data to minimize cost or burden. In this paper, we develop active learning methods that are tailored to the mHealth and HAR domains to address the problems of labeled data scarcity and the cost of labeled data collection. Specifically, we leverage between-user similarity to propose a novel hierarchical active learning framework that personalizes models for each user while sharing the labeled data collection burden across a group, thereby reducing the labeling effort required by any individual user. We evaluate our framework on a publicly available human activity recognition dataset. Our hierarchical active learning framework on average achieves between a 20% and 70% reduction in labeling effort when compared to standard active learning methods.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Prior work in mobile health has demonstrated that personalized models trained using data from the individual they are applied to often perform better than global, non-personalized models [1], [2]. This performance gap results from the fact that there can be substantial and systematic differences between individuals in terms of how they perform actions like smoking or eating or how their physiology reflects behavioral states like stress. While off-the-shelf wearable technology can be readily deployed leading to an abundance of unlabeled data, developing personalized models require access to labeled examples. The collection of labels typically results in higher cost or higher burden on the user depending on how labeled data are collected (e.g., in the lab setting via direct observation versus provided by the user via self report).

In this paper, we investigate machine learning methods to minimize the labeling effort required to learn personalized models. Specifically, we develop an approach that combines aspects of transfer learning, hierarchical clustering, and active learning to personalize a base model to every individual in a group while minimizing the labeling burden by selectively sharing the labeling effort across similar users in the group. We evaluate the proposed hierarchical active learning methods in simulation using an existing human activity recognition (HAR) data set. We develop methods in the pool-based setting as a proof of concept that hierarchical active learning methods can lead to significant reductions in the volume of labeled data needed to effectively personalize activity recognition models relative to existing active learning methods. Indeed, our results show between a 20% and 70% reduction in labeling effort when compared to standard active learning methods.

This work was partially supported by the National Institutes of Health award 1U54EB020404 and the National Science Foundation awards IIS-1350522 and IIS-1722792.

II. Background and Related Work

Our proposed hierarchical active learning framework leverages several components including a base classifier, a transfer learning approach, and a hierarchical clustering approach. We briefly review these components below, along with baseline active learning methods.

A. Base Classifier

We use a standard binary logistic regression classifier with hand-engineered features in this work since it directly outputs class probabilities, which are needed by the active learning framework we propose. Given a feature vector $X \in \mathbb{R}^D$ consisting of D features, the binary logistic regression classifier returns the probability of that feature vector belonging to the positive class $P(Y = y|X = x) = \frac{1}{1+\exp(-y(b+W^T x))}$ where, W is a length D vector of feature weights, b is the bias term and $Y \in \{-1, +1\}$ represents the label for the instance X.

Given a dataset $\mathcal{D} = \{(y_n, x_n)\}_{n=1:N}$ of N labeled examples, the regularized maximum likelihood objective function used during learning is defined as:

$$\sum_{n=1}^{N} \log\left(1 + \exp(-y_n(b + W^T x_n))\right) + \lambda R(W) \qquad (1)$$

The first term is the log likelihood of N data examples and the second term is the regularizer with regularization hyper-parameter λ. In standard logistic regression, it is typical to regularize the weights W toward 0 using, for example, a two norm squared regularizer $R(W) = \lambda \|W\|_2^2$.

B. Transfer Learning Approach

In this work, we adopt a basic parameter transfer approach to model personalization [3]. We assume that a prior set of logistic regression model parameters W_p have been estimated from previously collected data in the same feature space. Given a small data set from a new individual, we learn parameters for the individual using the objective shown in the

previous section, but with a modified regularization function $R(W) = \lambda\|W - W_p\|_2^2$ that penalizes deviations from the previously estimated parameters. Given a new user with no data, the solution to this problem is $W = W_p$. As more data are collected for a new user, the estimated parameters can diverge more extensively from the prior parameters with the rate of personalization controlled by the λ hyper-parameter.

C. Hierarchical Clustering Approach

Hierarchical agglomerative clustering (HAC) is a type of unsupervised clustering algorithm that recursively merges clusters pairwise based on a linkage distance criterion [4]. In this work, we perform clustering on users versus individual data examples. The HAC algorithm begins by having access to unlabeled data examples from M users, in each iteration it merges a pair of users to form a cluster until all users are merged into a single cluster. HAC utilizes a similarity or distance metric to merge clusters, we provide more details on the similarity metric used in this work in Section III-A. The results of HAC are often organized and presented via a dendrogram. In our case, the leaf nodes in the dendrogram correspond to the original M users and the non-leaf nodes are clusters of users that result from the recursive merges.

D. Active Learning

Most prior work in active learning for wearable sensing concerns the human activity recognition task. Longstaff et al., propose pool-based active and semi-supervised learning techniques to collect labels [5]. Specifically, they used data from a between-subjects model as a base classifier and chose new examples to be added to the labeled set either using active learning or semi-supervised learning. The conclusion was that active learning performed better than other techniques only when there existed a performance gap when starting with a between-subjects model. Saeedi et al., perform collaborative active learning with a panel of experts rather than a single oracle [6]. Recent work also includes methods that combine both transfer and active learning into a single framework of active transfer learning [7]. Our approach merges ideas from transfer active learning with ideas from hierarchical learning to better share the labeling burden across a group of users while simultaneously respecting between user differences.

III. Hierarchical Active Learning

In this section, we describe our hierarchical active learning (HAL) model. The most straightforward approach to collecting labels for each user is to develop one active learning model per user like shown in Figure 1a. In this approach, we can potentially reduce the number of labels needed by warm starting active learning using parameter transfer from a source model (denoted by SRC in the Figure 1a). The drawback of this approach is it does not leverage the similarities between users, which means further reductions in labeling effort may be possible depending on how similar the different users are. The other extreme is to share all examples between all users, but this approach is intrinsically limited as well unless the users are all highly similar.

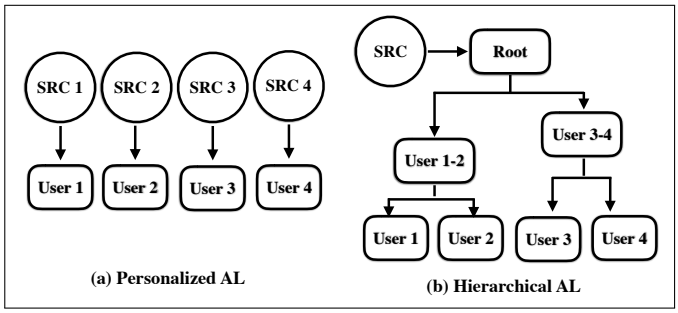

Fig. 1. Variants of active learning on a dataset with four users. (a) personalized active learning (b) hierarchical active learning. Here SRC refers to the source domain model.

An alternate approach is to group similar users into clusters and to share labels within cluster. The main questions are then how to assess the similarity between users and how to select an optimal number of clusters. Our proposed approach side-steps the problem of selecting a single clustering by using a hierarchical clustering and sharing information from labeled data using a hierarchical application of parameter transfer. Our objective is to apply active learning to improve the overall performance for all users simultaneously using the proposed hierarchical model structure to share information across user. We describe our clustering approach and active learning approach in detail in the next two sections.

A. Learning the hierarchical structure

Our approach to learn a hierarchical clustering over users is based on HAC as described in Section II-C. For the similarity function between users, we train an auxiliary classifier to attempt to discriminate between unlabeled examples for a given pair of users. We train such a model for each pair of users and use the classification error scores that result as the similarity scores. The intuition is that if a classifier can not distinguish between unlabeled data cases from two users, then the users are similar. This approach only requires unlabeled data, is very robust, easy to compute, minimally sensitive to outliers and no additional hyperparameters are introduced. The results of HAC are presented in a dendrogram as illustrated in Figure 1b. In this example, users 1 and 2 are most similar and hence are merged first followed by users 3 and 4 and the last merge happens at the root.

B. Transfer Active learning in the hierarchical structure

We leverage the dendrogram created in the previous step as a data structure to perform hierarchical transfer active learning. Specifically, we create one classifier for every node in the dendrogram using its parent to provide the prior model (the root node transfers from a prior source model). The leaf nodes correspond to single users and a model is learned for each leaf using data for that user only and its parent's model as a regularizer. The model at each internal node uses as training examples the labeled examples of all of its descendants that

are leaf nodes and its parent's model as a regularizer. There are thus two forms of labeled data sharing in the model: direct sharing within node and soft, indirect sharing via the hierarchical regularizer.

We perform active learning over all users (*i.e. leaf nodes*) in a round robin fashion from left-to-right. We use each user's current model in turn to compute the entropy over that user's unlabeled pool of examples. We select the example with maximum entropy and add it to that user's labeled data set. We then re-train all of the models in the hierarchy to take the new labeled example into account. This is a modification of the classical uncertainty sampling approach to active learning.

IV. Empirical Protocols

In this section, we discuss the empirical protocols and experimental details used to generate results in the next section.

A. Dataset

We use a publicly available human activity recognition dataset with 60 users, 300K minutes of data and about 116 reported activity types [8]. The study users wore a smartwatch and carried a smartphone. We use a version of the dataset with features computed over one minute windows of sensor data. In total, 175 features are available and organized into five groups including smartphone accelerometer and gyroscope (52), smartwatch accelerometer and gyroscope (46), microphone (26), location information (17) and features pertaining to phone status (34). Users provided activity labels via an app running on the smartphone.

In our experiments, we focus on three activities with positive labels from a wide set of users: "Sleep", "Drive," and "Surf Internet." In Table I, we provide the number of users, number of positive and negative examples, and the best reported performance in a binary classification setting for each activity.

B. Baseline Methods

We compare the performance of active learning approaches to two baseline methods. For both baselines, we use regularized logistic regression as the base model and perform hyperparameter selection using nested stratified 5-fold cross validation.

Within-User: This follows a straight within-user evaluation protocol. We train a prediction model on $k-1$ folds and evaluate on the held out k^{th} fold for each user.

Between-User: This follows the leave-one-user-out evaluation protocol. We train a prediction model on data from $M-1$ users and test on the held out M^{th} user. We use ten labeled examples (five positive and five negative) uniformly sampled at random from the $M-1$ users to simulate label scarcity in the between user model, which we then use as the source model for transfer learning.

C. Active Learning Methods

We use penalized logistic regression with transfer as the prediction model for all active learning methods. We contrast two different active learning strategies described in detail below: Personalized Active Learning (PAL) and Hierarchical Active Learning (HAL).

Personalized Active Learning (PAL): This is the standard version of active learning where we develop one active learning model per user. For each user, we randomly partition the data samples into k stratified folds ($k = 5$). For each user we use the data from $k-1$ folds as the sample pool and test on the held out k^{th} fold. We use the between-user model parameters as prior model parameters (W_p) that we transfer from. We train this prior model using a proxy dataset which consists of five positive and five negative examples sampled uniformly at random from the M users. Importantly, we remove these ten labeled examples from the respective sample pools so that they are not reused during active learning. We use uncertainty sampling with entropy utility as the active learning query selection approach. The prediction model is retrained after each query using only the actively learned examples.

PAL models start with a regularization parameters transferred from the prior model that is re-tuned after every 20 iterations during active learning. During retuning, we perform 5-fold cross validation on actively labeled examples to pick the best penalty from a range of $1e^{-4}$ to $1e^{+4}$. This retuning is triggered only when there are at least five positive and five negative actively learned examples. We perform PAL for each target activity for a total fixed budget of 100 labeled examples per user. This process is completely independent across all users with no sharing of information except for the common source model used for transfer learning.

Hierarchical Active Learning (HAL): We perform HAL over all users in a round robin fashion. The prior model is constructed as for the PAL approach. All models are initialized to the prior model. On each active learning round, we use the current model for the current user to select an unlabeled instance to query from that user's unlabeled pool. As with PAL, we use entropy-based uncertainty sampling as the active learning query selection method. Once a label is obtained, we re-train all models in the hierarchy. When re-training prediction prediction models at each node in the dendrogram, we use the model from the immediate parent node as a prior model (W_p) (the root node uses the source domain model (SRC) as its prior model).

HAL models start with a regularization parameter transferred from the SRC model and are re-tuned after every M^{th} iteration during active learning. M is the number of users in the data set. We perform HAL for each target activity using a total budget of $M \times B_T$ labeled examples where, B_T is the budget for target activity T listed as "# labels/user" in Table I. The total number of models to be updated after each query is $M + M - 1$. For each user, we evaluate the prediction model from that user's leaf node on that user's respective test set. All other details are similar to PAL.

D. Hierarchical agglomerative clustering

We perform HAC using a precomputed between-user similarity matrix. For each pair of users we compute the similarity score as the balanced accuracy of a model that attempts to discriminate between unlabeled data from each pair of users. We perform a stratified 5-fold cross validation to learn

TABLE I
DATASET STATISTICS FOR THREE ACTIVITIES ALONG WITH BASELINE, PAL AND HAL PERFORMANCE.

Activity	# Users	# Positive	# Negative	Between User	PAL Perf.	PAL # labels/user	HAL Perf.	HAL # labels/user	Within User	Best Prev. Reported
Sleep	38	42955	134045	0.77±0.01	0.88±0.01	100	0.87±0.01	20	0.91±0.01	0.89
Drive	24	5034	171966	0.74±0.01	0.82±0.02	100	0.82±0.02	60	0.87±0.01	0.87
Surf Internet	28	11641	165359	0.50±0.01	0.61±0.03	100	0.59±0.03	40	0.75±0.02	0.63

the required discriminative models and compute the mean balanced accuracy.

E. Evaluation Metrics and Reporting Results

Due the sample imbalances, we report balanced accuracy (BA) as in [8] as $\frac{1}{2}(TPR + TNR)$ where, TPR and TNR are true positive rate and true negative rates respectively. This metric ranges between 0 to 1 with greater balanced accuracy score indicating better predictive performance. We repeat each data analysis five times with different random seeds to average over the effects of different train/test partitions and resulting querying sequences. We compute the balanced accuracy per user as a mean over five repetitions and five folds. In the results section, we only report the mean balanced accuracy over users along with standard error of the mean over users.

V. RESULTS AND DISCUSSION

We report the average balanced accuracy across users along with the standard error of the mean for the selected activities in Table I. We view the within and between-user protocols as two extremes of access of labeled examples and personalization. At one end, the within-user protocol has access to large quantities of labeled examples ($\sim 80\%$) from each user to personalize the model before making predictions. This represents a best case scenario in terms of personalization, but would require a high cost in terms of user burden. At the other end is the between-user protocol which has no access to labeled examples for personalization. There exists a performance gap of between 5% to 15% between the two baseline models over all activities that we aim to bridge via active learning methods.[1]

For both PAL and HAL, we first note that both approaches consistently out-perform the use of random queries instead of uncertainty sampling. We thus focus on contrasting these approaches against each other. As we can see the performance gap between between HAL and PAL is within one standard error, but HAL can achieve this result using a much lower number of queries per user. On the "Sleep" task, HAL matches PAL using only 20 queries per user versus PAL's 100. Similarly on the "Drive" and "Surf Internet" tasks, HAL uses 60 and 40 queries per user to match PAL's performance with 100 queries per user. These are substantial reductions in the volume of labeled data needed to approach the performance of the best-case within-subject models.

[1]There also exists some differences between the within-user performance and the best reported within-user performance from [8]. This can be attributed to differences in user inclusion criterion that we enforced in our experiments.

Further, we observed that while HAL starts at a lower performance than PAL on average, it quickly surpasses it as the number of queries increases. The cross-over for sleep activity happens with as few as 50 labeled examples, which translates to each user labeling about a minute and half of their sensor data in the HAL setting due to the sharing of labels within the hierarchy. We observe similar trends for other activities as well, but the cross over happens at higher numbers of total queries. This indicates that HAL has the ability to produce better models than PAL in the early stages of active learning as well.

To conclude, we have demonstrated that our proposed HAL framework can achieve comparable performance to PAL while requiring many fewer labels. Future work for this approach includes moving it from the off-line pool-based setting, to the more realistic real-time streaming setting where it can be deployed in a live user study, as well as relaxing standard assumptions like the assumption that the user always responds to issued queries.

REFERENCES

[1] A. A. Ali, S. M. Hossain, K. Hovsepian, M. M. Rahman, K. Plarre, and S. Kumar, "mpuff: automated detection of cigarette smoking puffs from respiration measurements," in *Proceedings of the 11th international conference on Information Processing in Sensor Networks*. ACM, 2012, pp. 269–280.

[2] A. Natarajan, E. Gaiser, G. Angarita, R. Malison, D. Ganesan, and B. Marlin, "Conditional random fields for morphological analysis of wireless ecg signals," in *Proceedings of the 5th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*. ACM, 2014, pp. 370–379.

[3] C. Chelba and A. Acero, "Adaptation of maximum entropy capitalizer: Little data can help a lot," *Computer Speech & Language*, vol. 20, no. 4, pp. 382–399, 2006.

[4] L. Rocach and O. Maimon, "Clustering methods data mining and knowledge discovery handbook," *Springer US*, p. 321, 2005.

[5] B. Longstaff, S. Reddy, and D. Estrin, "Improving activity classification for health applications on mobile devices using active and semi-supervised learning," in *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2010 4th International Conference on-NO PERMISSIONS*. IEEE, 2010, pp. 1–7.

[6] R. Saeedi, K. Sasani, and A. H. Gebremedhin, "Co-meal: Cost-optimal multi-expert active learning architecture for mobile health monitoring," in *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. ACM, 2017, pp. 432–441.

[7] X. Wang, T.-K. Huang, and J. Schneider, "Active transfer learning under model shift," in *International Conference on Machine Learning*, 2014, pp. 1305–1313.

[8] 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.