



<http://www.diva-portal.org>

Preprint

This is the submitted version of a paper presented at *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*.

Citation for the original published paper:

Roy, D., Girdzijauskas, S. (2023)

Temporal Differential Privacy for Human Activity Recognition

In: (pp. 1-10-).

<https://doi.org/10.1109/DSAA60987.2023.10302475>

N.B. When citing this work, cite the original published paper.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-339761>

# Temporal Differential Privacy for Human Activity Recognition

Debaditya Roy

Royal Institute of Technology (KTH)  
Qamcom Research and Technology AB  
Stockholm, Sweden  
droy@kth.se

Šarūnas Girdzijauskas

Royal Institute of Technology (KTH)  
Research Institutes of Sweden (RISE)  
Stockholm, Sweden  
sarunasg@kth.se

**Abstract**—Differential privacy (DP) is a method to protect individual privacy when the data is used for downstream analytical tasks. The core ability of DP to quantify privacy numerically separates it from other privacy-preserving methods. In human activity recognition (HAR), differential privacy can protect users’ privacy who contribute their data to train machine learning algorithms. While some methods are developed for privacy protection in such cases, no method quantifies privacy and seamlessly integrates into machine learning frameworks like DP. The paper proposes a DP framework called *TEMPDIFF* (short for temporal differential privacy), which guarantees privacy preserving human activity recognition for wearable time-series data with competitive classification performance and works with any machine-learning/deep-learning methods. *TEMPDIFF* capitalizes on the temporal characteristics of wearable sensor data to improve the modelling task, which enhances the privacy-utility tradeoff. *TEMPDIFF* uses ensembling and a novel *temporal partitioning* algorithm for time-series data to ensure optimal training of ensemble models. In *TEMPDIFF*, consensus through ensembling and the addition of controlled Laplacian noise obscures sensitive information used to train the models, guaranteeing strict levels of differential privacy. The proposed method is evaluated on two popular HAR datasets. It outperforms the classification accuracy and privacy budget for both datasets compared to the state-of-the-art approaches.

**Index Terms**—Privacy, Differential Privacy, Deep Learning, Time-Series, Human Activity Recognition

## I. INTRODUCTION

Differential privacy (DP) is a privacy-preserving technique that can protect individuals’ privacy when their data is used for machine learning or other data analysis tasks [6]–[8]. The idea is to add controlled noise to the data so that information about individuals gets abstracted. In this way, the attacker cannot identify individual information, but the aggregated data can still be successfully used for downstream analytical tasks. One potential use of differential privacy is in the area of human activity recognition (HAR). In HAR, differential privacy can be used to protect the privacy of individuals whose data is used to train machine learning models for activity recognition. For

This project has received funding from the European Union’s Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 813162. This paper’s content reflects only the views of its author(s). The European Commission/ Research Executive Agency is not responsible for any use that may be made of the information it contains. We would also like to thank Qamcom Research and Technology for funding the project.

example, consider a scenario where a company is developing a machine-learning model to recognize and classify different types of physical activity, such as running, walking, or biking. This model might be trained on data collected from many individuals’ sensors or devices. Without differential privacy, the training data for this model could potentially be used to reveal sensitive information about the individuals who provided the data, such as their daily routines, physical activity levels, and location. By applying differential privacy to the training data, it is possible to protect the privacy of these individuals while still allowing the model to be trained on a large and diverse dataset. Thus improving both privacy and utility in the downstream HAR task.

In promoting the utility of HAR tasks, ensembles have played a significant role by improving their classification performance [3], [5], [15], [22], [27]. Moreover, ensembles also promote privacy through DP techniques such as *subsampling and aggregation* [8]. The approach involves training an ensemble of models on disjoint data partitions and aggregating them with noise to achieve differential privacy [19], [22], [23], [32]. The ensembling method in this strategy relies on the idea that if multiple models, trained on disjoint data subsets, agree on a classification, they have learned a general trend rather than memorizing specific examples. Consequently, even though highly sensitive data might influence a prediction’s outcome in this setup, it will only be present in a single data subset and train a single model. As such, this data would only contribute one input to the ensemble’s aggregation function, making it inherently challenging to identify its source solely from the aggregated output. A larger number of models influencing the aggregation function enhances the capacity to obscure private information. The simple proposition of enhanced privacy through more models makes *subsampling and aggregation* a rather convenient choice for integrating DP in machine learning problems. However, to the best of our knowledge, no previous works have attempted to formulate a differentially private HAR system using the *subsampling and aggregation*. Hence in this work, we employ the same to train a large ensemble of models to provide differentially private HAR for time-series data.

Realizing the above goal for time-series data requires the following:

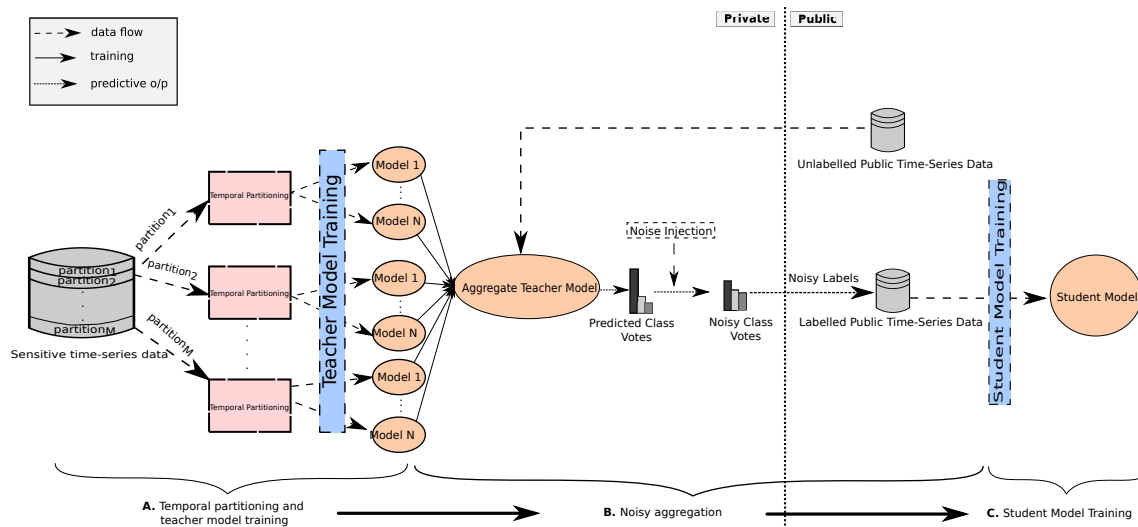


Fig. 1. *TEMPDIFF* for differentially private HAR. **A.** We partition our sensitive time-series HAR data in  $M$  partitions and feed each partition in *temporal partitioning*, which trains  $N$  teacher models per partition. **B.** Public time-series HAR data is labeled using the trained teachers and noisy aggregation. **C.** Train a student model using the public time-series data and noisy labels.

- Partitioning the time-series data into disjoint partitions.
- Extracting temporal sequences of specific window sizes from each partition to train individual models for aggregation.

However, training many models necessitate a corresponding increase in data partitions. As the number of partitions grows, the amount of training data per partition decreases, potentially compromising the quality of model training. While the quantity of the training data can be counteracted by having more meaningful and diverse data, using a single window size to extract temporal sequences in the time-series classification limits the diversity. Instead, using multiple window sizes to extract different temporal sequences offers a broadened exploration of the time series by presenting a different view of the data. Leveraging these temporal sequences to train different models improves classification over the standard method. Moreover, the efficacy of this technique, employing varying window sizes for enhanced data representation, has also been validated in prior HAR studies [21], [27]. Hence in this paper, we propose *TEMPDIFF* (temporal differential privacy), an ensembling framework suited for time-series-based HAR that incorporates the temporality of the data to improve model training and provides differentially private predictions with competitive utility. The overview of *TEMPDIFF* is presented in Figure 1. We assume the HAR data used to train the models are sensitive and private. Firstly, the sensitive data is partitioned into multiple unique subsets (similar to the *subsampling* strategy), following which we further partition each subset based on different time windows (*Temporal Partitioning* box in Figure 1). The *Temporal Partitioning* leads to multiple sub-partitions from each subset with unique temporal properties. These sub-partitions are used to train individual models. In contrast to default ensembling-based methods, our approach allows better exploration of the time-series data in each partition.

As a consequence, the models are appropriately trained. The consensus from the trained models (on the classification task) during *aggregation* helps obscure information about individual data items and improve HAR classification.

To make a single private prediction in *TEMPDIFF*, individual models' outcomes are aggregated and injected with Laplacian noise [8]. However, using the models directly for making a series of predictions incurs a privacy loss for every prediction [7], [23]. To bypass this, we utilize the *knowledge distillation* approach previously used in differential privacy literature [4], [13], [20], [22], [23]. In this setup, we call the ensemble of trained models as *teacher models*. To reduce the privacy loss, the knowledge from the teacher ensembles is distilled into a *student model* through a limited number of non-sensitive queries using a public dataset. The public dataset (accessible by any adversary) is used on the *aggregate model* (Figure 1) to extract noisy labels. The noisy labels and the public data train the student model deployed for the downstream task. Similar architecture (called PATE) for non-time-series data was successfully demonstrated on non-temporal settings ([22], [23]). However, in our work, the *temporal partitioning* allows seamless integration of *TEMPDIFF* for time-series-based HAR problems. On HAR workloads, *TEMPDIFF* provides stronger privacy guarantees and better-distilled student model classification performance than PATE. *TEMPDIFF* do not have any assumptions in the type of model used for training. In our work, we have used *LSTM* networks for modeling since they are efficient for time-series data. To measure differential privacy, we have used the adaptation *Moment's Accountant Technique*, introduced by Abadi et al. [1] and subsequently used for an ensemble set up by Papernot et al. [22], [23]. It allows us to calculate the privacy cost/budget  $\epsilon$  (less is better for privacy) of an individual query to the teacher models for an input noise. The privacy cost can be composed over multiple

queries to formulate the system’s total cost as  $(\epsilon, \delta)$  differential privacy.  $\delta$  measures the probability that a differentially private algorithm will fail to preserve privacy.

The main contributions of the paper are as follows:

- We propose *TEMPDIFF*, a framework that guarantees a well-balanced tradeoff between privacy and utility (classification performance) for time-series-based HAR workloads. While some previous works connect differential privacy and HAR, to the best of our knowledge, this is the first work that approaches it with the privacy-utility tradeoff on public HAR datasets.
- We improve upon the previous PATE baseline introduced by Papernot et al. [22] for time-series-based HAR datasets. In particular, our *temporal partitioning* strategy allows the creation of unique and informative training data from a particular subset of partitioned data. This improves the teacher model training and hence the overall classification.
- To show the method’s effectiveness, we extensively evaluate the framework on two popular HAR datasets (WISDM and PAMAP2). For both datasets, the proposed framework outperforms the state-of-the-art (*PATE* [22]) in classification accuracy and privacy. On the WISDM and PAMAP2 datasets, our framework achieved a  $(\epsilon, \delta)$  privacy bound of  $(1.4, 10^{-5})$  and  $(8, 10^{-5})$  for achieving a 0.82 and 0.79 accuracy respectively. On comparison, the baseline *PATE* achieved  $(2, 10^{-5})$  for an accuracy of 0.74 on the WISDM dataset. For the PAMPA2 dataset, *PATE* achieved  $(33, 10^{-5})$  for an accuracy of 0.66. We also demonstrate the different tradeoffs between performance, privacy, and the number of teacher models for this framework.

The paper is organized as follows: The next section looks at the *related works*, followed by *methods*. Then, we dive into an *extensive evaluation*, concluding with *conclusion and future works*.

## II. RELATED WORKS

Human activity recognition (HAR) is a widely researched problem with many practical applications. HAR algorithms have been extensively deployed in different sectors, such as sports [28], activity tracking [24], healthcare [2], etc. The extraction of temporal sequences is popularly used in many previous HAR works [11], [16], [27]. However, none of the above works were designed to handle privacy. A more focused effort towards incorporating privacy into HAR problems has been undertaken recently. Sozinov et al. [30] developed a federated learning algorithm for HAR, where the data is private to individual users. Kumari et al. [18] used homomorphic encryption to guarantee the security and privacy of activity tracker data for HAR workloads. However, these methods lack the privacy accountability of differential privacy.

Recently differential privacy has been integrated into different types of HAR problems [10], [21], [33]. In [21], the authors present a privacy-preserving model based on secure multi-party computation that makes a fully homomorphic

encryption multi-key. While this work is theoretically proved to be differentially private, it does not provide  $(\epsilon, \delta)$  results. It makes it hard to understand the privacy-utility tradeoff, a key feature of our framework. Zhang et al. [33] also provide a theoretically differentially private algorithm for HAR from channel state information of Wifi signals, but not the  $(\epsilon, \delta)$  numbers.

Differential privacy is a mathematically rigorous stream of privacy introduced by Dwork et al. [7]. Machine learning and deep learning algorithms have also been adapted for differential privacy and applied in different use cases. Shokri et al. proposed a differentially private SGD applied to deep learning-based methods [29]. In this method, they apply controlled noise to gradient updates to obscure the internal model parameters. Although this is still a popular method to achieve differential privacy, the complexity of the method is relatively high. Another differential privacy method, Private Aggregation of Teacher Ensembles (PATE), was proposed by Papernot et al. [22], [23]. They demonstrate a teacher-student-based ensemble framework with a better privacy-utility tradeoff than [29]. Our framework *TEMPDIFF* is inspired by the PATE framework. However, the data-partitioning strategy in our framework is specialized for time-series data and guarantees a better privacy-utility tradeoff compared to PATE.

## III. METHODS

### A. Problem Setting

This paper proposes a practical setup for achieving differential privacy in Human Activity Recognition tasks. In the setup, we assume that a significant portion of the user activity data is sensitive, requiring robust protective measures. This premise aligns with the reality that individuals generating data entrust only the data-collecting organization with their privacy assurance. Interestingly, unannotated and cheaply available public data can also be utilized to ensure the privacy of the proposed setup. We discuss how this is done in the ensuing subsections.

We aim to protect user data via differential privacy while ensuring utility simultaneously. In the process, we recognize the potential for two types of adversaries: black box and white box [12]. A black box adversary can access the deployed model and make unlimited queries, whereas a white box adversary can access the model’s internals. In a typical HAR environment, we frequently encounter black box adversaries attempting to infringe on privacy protections. In contrast, the white box adversaries are less common because companies seldom release their models. However, we can strengthen our privacy proposition if we argue that *TEMPDIFF* is secure from both adversaries.

### B. *TEMPDIFF*

Our framework, *TEMPDIFF*, is proposed to enhance the privacy/utility tradeoff in time-series-based Human Activity Recognition (HAR) problems. Figure 1 depicts the framework. *TEMPDIFF* operates in three distinct stages:

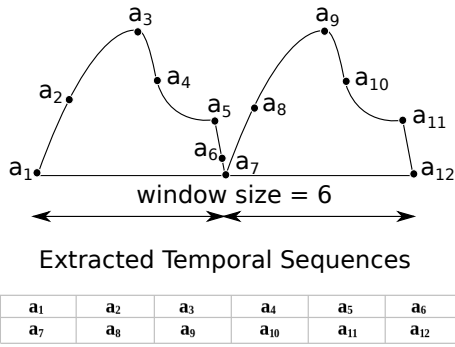


Fig. 2. Extracting temporal sequences from time-series with a particular window size.

- (A) *Temporal partitioning and teacher model training*: During this stage, the input dataset is divided into  $M$  disjoint subsets, each of which is then passed to the *temporal partitioning* module. This further divides each subset into  $N$  sub-partitions. An individual teacher model is trained for each of these sub-partitions. This results in  $M \times N$  teacher models derived from  $M$  original partitions.
- (B) *Noisy aggregation*: This stage focuses on making the aggregated teacher model predictions differentially private by injecting Laplacian noise [8]. In this stage, a small unlabelled public HAR dataset is labeled using the aggregated teacher models. Note that the labels obtained through the above process are noisy.
- (C) *Student model training*: The final stage involves training a student model using the noisy class labels and the public dataset. The model learns from the noisy class labels produced in the previous stage, thus distilling the knowledge of the teacher models while preserving privacy.

To ensure privacy, the three stages of *TEMPDIFF* operate in two parts, a *private part* and a *public part* (represented by the vertical line in Figure 1). The teacher models of the framework are trained with sensitive activity data originating from the users. Hence, they are within the *private part* of the framework to which an adversary has no access. The other part is the *public*, where the knowledge from the teacher models is distilled into a student model using a small public dataset. This student model is deployed for inference during HAR classification. Next, we discuss the three distinct stages of *TEMPDIFF* and how they are integrated to ensure privacy preservation.

1) *Temporal partitioning and teacher model training*: The distinguishing feature of the *TEMPDIFF* framework, which sets it apart from frameworks like PATE, is its *temporal partitioning* method. This technique efficiently segments the input time series, equipping each partition with a rich data set to facilitate training a unique teacher model, thereby creating a robust learner.

This approach promotes the growth of teacher models without compromising each model’s optimal performance. Conversely, using a traditional partitioning strategy to divide

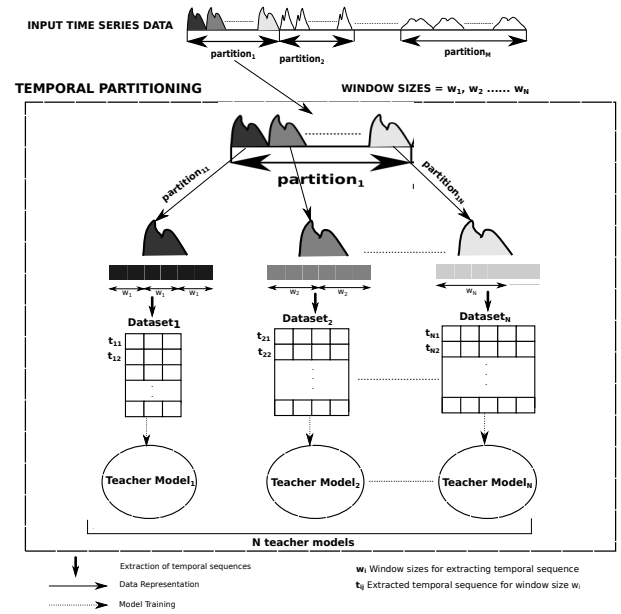


Fig. 3. Temporal partitioning and teacher model training in PATE. Here the method is demonstrated for a single partition of the full time-series data,  $partition_1$ .

the entire dataset into  $M$  partitions can result in each model only acquiring sub-optimal patterns as  $M$  increases. Given the limited data points in each partition with a growing  $M$ , a single teacher model might struggle to train adequately. This could lead to disagreement among the teacher models on test examples, negatively impacting classification performance and privacy. However, we can address this sub-optimality in model training by employing a more efficient partitioning strategy. The core idea behind the partitioning strategy involves curating different temporal representations from the time-series data. In particular, we can extract different temporal sequences from a raw time series using varying window sizes. The principle is illustrated in Figure 2 where a time-series data curve is denoted by a sequence of recordings  $a_1, a_2, \dots, a_{12}$ . Utilizing a window of size six, we extract the first six points and then slide the window to obtain the subsequent six points (here, we have non-overlapping windows). The sliding process creates two temporal sequences that form a dataset. This dataset can be utilized in the downstream pattern recognition task. By altering the window size, we can slide different-sized windows across the time series, extracting unique temporal sequences and corresponding datasets. This method has successfully been applied to HAR classification problems in previous state-of-the-art [23], [27]. Incorporating this concept into the *TEMPDIFF* framework enables the generation of multiple distinct datasets of temporal sequences from a single partition. Using those datasets, more teacher models can be trained optimally, improving the privacy and utility of the HAR task.

Figure 3 demonstrates the *temporal partitioning* method where the strategy is applied. In the figure, the input time series data is divided into  $M$  number of partitions, and there are  $N$  window sizes  $w_1, w_2, \dots, w_N$ . The *temporal*

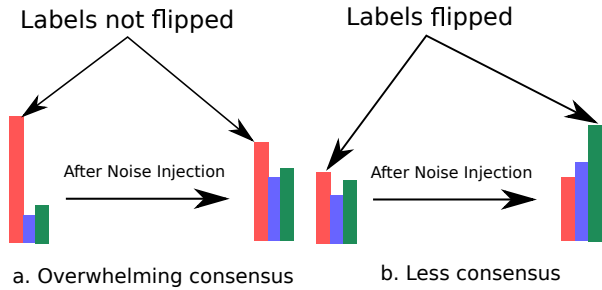


Fig. 4. Class votes after aggregating teacher models with a) overwhelming consensus and b) lesser consensus.

*partitioning* module in the figure is shown on  $partition_1$ . Based on  $N$ , we further divide  $partition_1$  into unique sub-partitions ( $partition_{11}, \dots, partition_{1N}$  in Figure 3). The time-series data from  $partition_1$  is represented with three distinct colors to represent these sub-partitions. Each window size,  $w_1, \dots, w_N$ , is slid over each sub-partition  $partition_{11}, \dots, partition_{1N}$  respectively. E.g. the sliding of  $w_1$  on  $partition_{11}$  forms temporal sequences  $t_{11}, t_{12}, \dots$  etc. Each of these slides forms a row of the  $Dataset_1$ . Similarly,  $Dataset_2$  to  $Dataset_N$  is constructed with  $w_2, \dots, w_N$ . Thus, we extract  $N$  datasets for every partition using the time windows  $w_1, w_2, \dots, w_N$ . These  $N$  datasets are used to train  $N$  teacher models per partition, and a total  $M \times N$  teacher models are trained for  $M$  partitions. While our approach also involves the division of the dataset into  $M \times N$  disjoint partitions, similar to the PATE methodology, our framework significantly diverges due to the inclusion of temporality. This unique incorporation allows for more robust representations of teacher models, thereby offering an improvement over the conventional PATE approach. The algorithm for temporal partitioning is presented in Appendix C Algorithm 1<sup>1</sup>.

2) *Noisy Aggregation*: To make the outcomes of the teacher models differentially private, we must aggregate the outcomes and add Laplacian noise [13]. While making predictions on a dataset, the outputs of each of these teacher models participate in voting. After they vote for a particular class, the total number of votes per class is aggregated, and Laplacian noise with scale  $\frac{1}{\gamma}$  (where  $\gamma$  is called the privacy parameter) is added to the class votes. The equation below quantifies it,

$$output = \operatorname{argmax}[vote + \operatorname{Lap}(\frac{1}{\gamma})]$$

The *vote* (in the above equation) is the aggregate votes assigned per class by all the teacher models. Using the Laplacian mechanism on the aggregated votes makes the outcomes  $(2\gamma, 0)$ -DP. Thus, lower values of  $\gamma$  should provide strong privacy guarantees. However, lower  $\gamma$  also means high noise to the votes that might hamper the classification result.

Securing a strong agreement among teacher models is crucial in balancing the privacy-utility tradeoff. It permits the addition of increased noise to the teacher votes without

flipping the label. This concept is visually depicted in Figure 4, showing the class votes after aggregating the teacher models. The figure illustrates two scenarios—one where teacher model consensus is overwhelmingly high and another where consensus is comparatively lower. In the case of strong consensus, introducing substantial noise alters class votes but does not affect the overall outcome. However, adding heavy noise can entirely reverse the outcome in the scenario with weaker consensus. This shift can detrimentally affect downstream classification performance. To prevent this in the second case, we would need to limit the amount of noise added, compromising privacy. Thus, achieving teacher consensus is vital for balancing classification performance and privacy preservation.

3) *Student Model Training*: While deploying our teacher models through aggregation may yield differentially private predictions, each query to this model incurs a certain degree of privacy loss. Consequently, after a specific number of queries, the model loses its privacy protections, rendering user data vulnerable to black-box adversaries. Therefore, it is critical to constrain the overall privacy loss through queries.

Adopting a teacher-student architecture can effectively limit privacy loss, allowing unlimited queries to a differentially private system without privacy degradation. In this framework, inaccessible private data is used to train multiple teacher models (as depicted in the *private* box of Figure 1). The noisy aggregation of these models guarantees privacy-preserving predictions. However, rather than deploying the teachers directly, their knowledge is distilled into a student model. This process involves using teacher models to infer noisy labels of a small public dataset (as seen in the *public* part of Figure 1). Paired with the public data, these labels are then used to train a student model, which is subsequently deployed to predict human activities. Through this teacher-student setup, the total privacy loss is upper bound by the loss incurred through limited queries from the small public dataset (to the aggregated teacher models).

### C. Protection against adversaries

The teacher-student knowledge distillation offers protection from both black-box and white-box adversaries. Through the deployment of the student model, we upper bound the total privacy loss. Hence, no amount of queries a black box adversary makes can incur additional privacy loss. Furthermore, a white-box adversary with access to the student model's internal parameters can reverse engineer the noisy labels obtained from the teacher models. Nonetheless, the original sensitive training data remains secure from such adversarial attacks, as it is not used to train the student model. If, instead, we deployed the teacher model directly, we would have failed against both adversaries.

### D. Privacy Analysis

1) *Differential Privacy*: Differential privacy is an industry-standard that can be used to analyze the privacy of algorithms operating on datasets. The algorithms can also be machine learning or deep learning algorithms. Formally differential

<sup>1</sup>[http://bit.ly/dsaa\\_dphar](http://bit.ly/dsaa_dphar)

privacy can be defined as follows: A randomized algorithm (or mechanism)  $M$  with domain  $D$  and range  $R$  satisfies  $(\epsilon, \delta)$  differential privacy if for any two adjacent inputs from two databases (that differs by one record)  $x, y \in D$  and for any subset of outputs  $S \in R$  it holds that:

$$P[M(x) \in S] \leq e^\epsilon P[M(y) \in S] + \delta$$

Here  $\delta$  is the probability of failure of differential privacy. The definition of differential privacy above is adopted from Dwork et al. [8]. Every query to a differentially private algorithm leaks privacy and hence incurs a privacy loss. This privacy loss can be composed of multiple such queries. Hence, the algorithm is no longer differentially private after a bounded number of queries.

Privacy loss and its random variable are two entities derived as a difference of probability distribution from running  $x$  and  $y$  on  $M$ . Privacy loss for an outcome  $o$  and input  $a$  is defined as,

$$c(o; M, a, x, y) \triangleq \ln \frac{P[M(a, x) = o]}{P[M(a, y) = o]} \quad (1)$$

It is not hard to see that the above definition of privacy loss can be drawn from the definition of differential privacy. The privacy loss random variable is defined as,

$$C(M, a, x, y) \triangleq c(M(x); M, a, x, y) \quad (2)$$

i.e., the random variable defined by evaluating privacy loss at an outcome obtained from  $M(x)$ . The privacy loss per query can be aggregated over the total number of queries naively, i.e., simple summation, or through strong composition theorem [9]. The latter provides better bounds on the overall privacy of the system and forms the basis of privacy accounting in many differential privacy systems. However, it was shown by Abadi et al. that the strong composition theorem can also provide loose guarantees. To mitigate this, Abadi et al. introduced a technique known as the Moment's Accountant [1] that keeps track of the privacy budget in different privacy analysis frameworks, e.g., PATE [22]. In the following subsection, we discuss the specifics of Moment's Accountant Technique and show how it can be used to calculate the privacy budget of *TEMPDIFF*.

2) *Privacy Accounting*: The core of the Moment's Accountant technique revolves around estimating the moments of the privacy loss random variable denoted at the moment  $\lambda$  as,

$$\alpha_M(\lambda) \triangleq \max_{a, x, y} \alpha_M(\lambda; a, x, y) \quad (3)$$

where  $\alpha_M(\lambda; a, x, y) \triangleq \ln E[e^{\lambda C(M, a, x, y)}]$  is the moment generating function of the privacy loss random variable, i.e.,  $C(M, a, x, y)$ . The Moment's Accountant keeps track of  $\alpha_M(\lambda)$  for  $\lambda = 1, 2, \dots, k$  for some chosen  $k$ . These tracked values are then used to calculate an upper bound on the total privacy cost. The essence of Moment's Accountant technique lies in its insight that the moments of the privacy loss random variable offer extensive information about the distribution of privacy loss and the total privacy budget. Consequently, it provides a much more accurate estimate of the total privacy

cost than the strong composition theorem. There are three steps to obtain the privacy budget  $\epsilon$  through Moment's Accountant method.

- 1) Calculate moment generating functions for mechanism  $M$  and the moment of privacy loss using equation (3) for a moment  $\lambda$ .
- 2) Compose the moments calculated using equation (3) using the below equation, where a mechanism  $M$  comprises a series of adaptive mechanisms  $M_i$  where  $i = 1, 2, \dots, k$ .

$$\alpha_M(\lambda) = \sum_{i=1}^k \alpha_{M_i}(\lambda) \quad (4)$$

- 3) **Theorem:** For any  $\epsilon$ , a given mechanism  $M$  is  $(\epsilon, \delta)$ -DP for,

$$\delta = \min_{\lambda} \exp(\alpha_M(\lambda) - \lambda \epsilon) \quad (5)$$

Solve the equation given by the theorem above for a particular  $\delta$  to obtain the  $\epsilon$  either through a closed-form solution or by searching over a moment space of  $\lambda_1, \lambda_2, \dots$ , etc.

The above equations' proofs are detailed in the paper by Abadi et al. [1]. The privacy budget for differentially private algorithms can be calculated using the above three steps. The choice of the differentially private algorithm often offers flexibility in calculating a closed-form solution of the moments. In *TEMPDIFF*, we use the Laplacian mechanism during aggregation like in PATE by Papernot et al. [22] for differential privacy. Based on the findings by Dwork et al. [8], the Laplacian mechanism with a noise scale of  $\frac{1}{\gamma}$  is  $(2\gamma, 0)$ -DP. The closed-form solution of the moment-generating function for the Laplacian mechanism is given by,

$$\alpha(\lambda; a, x, y) \leq 2\gamma^2 \lambda(\lambda + 1) \quad (6)$$

When we apply the composition rule from equation (3) to the function above and then calculate the final privacy budget using equation (4), we can determine the data-independent privacy budget for the Laplacian aggregation mechanism. However, this data-independent analysis tends to provide a loose bound of privacy loss as it does not consider the agreement among models. Papernot et al. [22] introduced a data-dependent privacy analysis to address the above issue. Their approach assesses the degree of consensus amongst the ensemble of teacher models. The resultant effect is a reduction in the overall privacy cost when there is a high degree of agreement amongst these models. In ensemble learning, it has been noted that identifying the most similar data partitions for a given test sample becomes challenging when there is strong agreement between the teachers, even before the noise injection. This contributes to the fact that the sensitivity of a query decreases as the consensus among the teachers strengthens. Consequently, privacy loss associated with such outcomes is reduced. In this case, the moment of privacy loss,

$$\alpha(\lambda; a, x, y) \leq \ln\left((1 - q)\left(\frac{1 - q}{1 - e^{2\gamma q}}\right)^\lambda + qe^{2\gamma\lambda}\right) \quad (7)$$

, is satisfied by  $M$  a  $(2\gamma, 0)$ -DP mechanism, where  $q \geq P[M(x) \neq o^*]$  for an outcome  $o^*$ . Also,  $l, \gamma \geq 0$  and  $q \leq \frac{e^{2\gamma-1}}{e^{4\gamma-1}}$ . Furthermore,  $q$  can be upper bounded by

$$P[M(x) \neq j^*] \leq \sum_{j \neq j^*} \frac{2 + \gamma(n_{j^*} - n_j)}{4 \exp(\gamma(n_{j^*} - n_j))} \quad (8)$$

, where  $n$  is the label vector score for database  $x$  with  $n_{j^*} \geq n_j \forall j$ . The proof of the theorems that derive equation (7) and equation (8) is shown in [22]. Equations (7) and equation (8) help us calculate the data-dependent moment of privacy loss for given  $\lambda$ . The earlier steps involve calculating the final privacy budget of *TEMPDIFF*. We initially compute the smallest moments value between equations (7) and equation (8), considering a few specific  $\lambda$  values. Then, we apply equation (4) to compose these values and find  $\alpha_M(\lambda)$ . Using equation (5), we ultimately determine the  $(\epsilon, \delta)$  privacy guarantee for the given  $\alpha_M(\lambda)$ .

## IV. EVALUATION

### A. Datasets

For our experiments, we divide the dataset into three parts. This division is not equivalent to the partitioning method required for our algorithm but for training different parts of the framework and testing it.

- 1) *Training of teacher models* This data resides in the private part of the framework. It constitutes the maximum data split of the dataset.
- 2) *Training of student model* This portion of the data is the public data (accessible by any adversary) sent to the aggregated teacher ensemble for labeling. Then, a student model is trained on the public data and the aggregated noisy labels.
- 3) *Testing the student model* This is the partition of the data on which the student model is tested. The classification metrics are reported on this portion of the data.

We have used two datasets for our experiments: WISDM [31] and PAMAP2 [25]. The details of the dataset are presented in Appendix A<sup>1</sup>.

### B. Modelling architecture

LSTM-based neural network architectures have shown successful results in HAR tasks for the chosen datasets [11], [27]. Hence, we decided to stick to LSTM-based neural networks for modeling purposes. However, this framework is modeling choice agnostic; hence, any model can be used. The best hyperparameters for the models are provided in Appendix B<sup>1</sup> Table II, and the implementation details are provided in Appendix D<sup>1</sup>. The time-series classification also induces additional parameters for sequence length selection. This has been extensively explored in previous works. Hence we use window sizes from the previous literature [11], [14], [27].

### C. Experimental Setup

Our experiment aims to demonstrate the tradeoff between privacy and utility for *TEMPDIFF* and the vanilla version of *PATE* framework [22] (which we refer to as *Vanilla-PATE* in our experiments). We have different parameters that affect the functioning of our framework. Apart from the modelling parameters, we have three parameters that affect the privacy-utility tradeoff, namely *number of teacher models*, *window sizes* for parsing the time-series data, the *privacy parameter*,  $\gamma$ .

Based on our resources, we train up to 100 models for both *TEMPDIFF* and *Vanilla-PATE* for both datasets. In *Vanilla-PATE* to train  $M$  models, we make  $M$  partitions. For *TEMPDIFF*, we select a  $N$  window-sizes and  $M$  partitions to train  $M \times N$  teacher models. The details of window-size selection and estimation of the number of partitions for *TEMPDIFF* is shown in Appendix B<sup>1</sup>.

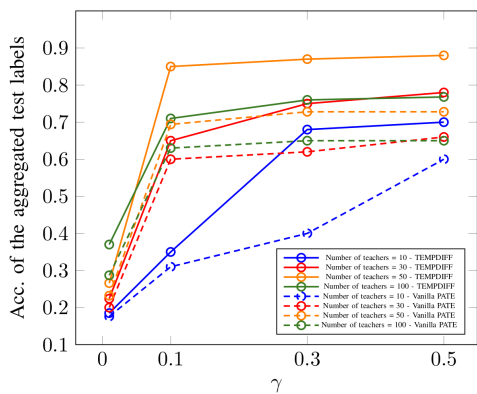
The privacy parameter  $\gamma$  affects the amount of Laplacian noise added to the outcome of the teacher models. We will test *TEMPDIFF* and *Vanilla-PATE* for each teacher model configuration using different values of privacy parameter  $\gamma$  (0.01, 0.1, 0.3, and 0.5) and a fixed value of  $\delta = 10^{-5}$ . The values of the  $\gamma, \delta$  are adapted from the previous work [22].

### D. Results

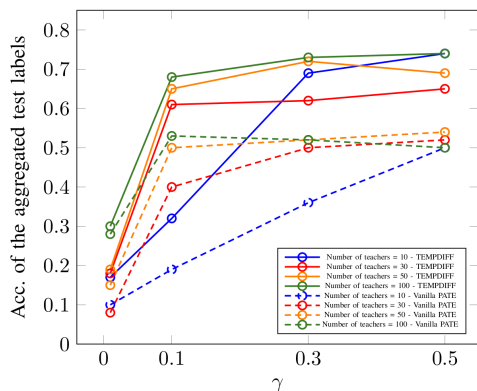
1) *Amount of noise that can be injected during the aggregation process*: In this analysis, we investigate the relationship between noise injection and the utility of the teacher models (see Figure 5). The  $x$ -axis represents  $\gamma$ , indicating the quantity of noise introduced to ensure differentially private aggregation. Given that the Laplacian mechanism renders each aggregation  $(2\gamma, 0)$ -DP, a smaller  $\gamma$  value implies enhanced privacy per label prediction. The  $y$ -axis represents the noisy aggregation accuracy on the test-set labels. Usually, when we have fewer teacher models, we observe that we trade off utility for strict privacy levels. This is evident in both *Vanilla-PATE* and *TEMPDIFF* across both datasets (as seen in Figure 5a and Figure 5b) when the number of teachers is 10. Conversely, more teacher models can effectively accommodate increased noise injection, thus strengthening privacy. E.g., in Figure 5a, when the number of teachers is 50 in *TEMPDIFF*, we get good accuracy even when the  $\gamma$  is low. Similar observations can be made for 100 teachers in the PAMAP2 dataset (Figure 5b).

Interestingly we observe that when the number of teachers is 50, we get the best privacy-utility tradeoff in *TEMPDIFF* for the WISDM dataset. On the other hand, we get the best privacy-utility tradeoff for *PAMAP2* using *TEMPDIFF* for 100 models (see Figure 5b). We suspect that the amount of data per partition in the WISDM dataset is insufficient for training 100 models; hence, the agreement among the models is low. This hampers the utility of the aggregated model.

Note that a gap exists between the aggregation accuracy among *TEMPDIFF* and *Vanilla-PATE*. This reinforces our argument that in *TEMPDIFF*, our *temporal partitioning* algorithm helps achieve better utility due to optimal training of the



(a) WISDM Dataset



(b) PAMAP2 Dataset

Fig. 5. Accuracy of the aggregated test set labels versus the inverse scale of the Laplacian noise  $\gamma$  grouped by the number of teachers (10, 30, 50, 100) for *TEMPDIFF* (solid lines) and *Vanilla-PATE* (dotted lines) for (a) WISDM dataset (top) and (b) PAMAP2 dataset (bottom).  $\gamma$  varies inversely to the injected noise and directly to the privacy per query.

TABLE I

PRIVACY-UTILITY TRADEOFF BETWEEN *TEMPDIFF*, *Vanilla-PATE* AND NON-PRIVATE STATE-OF-THE-ART BASELINE. NOTE THAT WE DO NOT HAVE ANY PRIVACY BUDGET ( $\epsilon$ ) FOR THE NON-PRIVATE BASELINE. *Number of queries* REPRESENT THE NUMBER OF EXAMPLES USED TO TRAIN THE PRIVATE MODELS. THE REPORTED ACCURACY IS ON THE STUDENT TEST SET.

Dataset	Method	Number of queries	Privacy Budget ( $\epsilon$ )	Accuracy	Non-private accuracy
WISDM	<i>TEMPDIFF</i>	1000	1.4	0.82	0.93 [27]
	<i>Vanilla-PATE</i>		2	0.74	
PAMAP2	<i>TEMPDIFF</i>	1600	8	0.79	0.88 [27]
	<i>Vanilla-PATE</i>		33	0.66	

teacher models. While, in *Vanilla-PATE*, most teacher models do not agree on their outcomes before the noisy aggregation (arising from sub-optimal training due to insufficient data). Thus, the noise injection changes the outcome so that the noisy label does not match the true label, lowering the accuracy.

2) *Privacy Budget and accuracy of the deployed student model*: In this subsection, we want to analyze the privacy-utility tradeoff grouped by the number of teacher models. In this setting, we plot the overall privacy budget and student model accuracy for each teacher model configuration for different noise injections for both *TEMPDIFF* and *Vanilla-*

*PATE*. The privacy budget in this analysis is composed across all queries, relying on the privacy analysis methodology we described previously.

The results of this analysis are plotted in Figure 6 and Figure 7. For this setting, the best privacy utility tradeoff would be represented by points with a low privacy budget and a high student accuracy, i.e., the scatter points on each graph's top left corners. The best privacy-utility tradeoff is observed for 50 models and  $\gamma = 0.5$  and  $\gamma = 0.3$  in the WISDM dataset. We have a comparable privacy budget for 100 models (Figure 6d), but the accuracy is lower. The highest noise injected with  $\gamma = 0.01$  for this dataset does not yield the best accuracies, even with many models indicating that this noise is too high to have any real utility. For the PAMAP2 dataset, good privacy-utility tradeoffs are offered by 100 teacher models and  $\gamma$  values of 0.1, 0.3, and 0.5. Here,  $\gamma = 0.01$  also adds too much noise to have any real utility.

### 3) Privacy-utility tradeoff comparison among baselines:

Table I compares the privacy-utility tradeoff for *TEMPDIFF* and *Vanilla-PATE* for both datasets. Please note that the reported accuracy of the private baselines is based on training our student model on a limited number of examples and their corresponding noisy labels (reported as **Number of queries** in Table I). We also report the non-private accuracy for each dataset from the state-of-the-art that uses the same test partitions [27]. The non-private baseline is trained on the fully available training data and evaluated on the same test partitions. It is seen that *TEMPDIFF* has a lower privacy budget compared to *Vanilla-PATE* for better accuracy in both datasets. As expected, the non-private baseline has the best classification accuracy.

4) *Impact of knowledge distillation*: One question for the proposed setting may arise is whether using only available public data to train the HAR model without *TEMPDIFF*'s teacher-student setup would yield equivalent classification performance. If so, it would imply that the privacy of the sensitive data is not a concern, as they are not used in training. In this case, simple training on the small publicly available training data and the subsequent model deployment is enough. Although we assume unannotated public data for a practical setup, we eliminate that assumption for this analysis. Hence in this experiment, we compare the performance of a model trained only on the public data to that of *TEMPDIFF*, where teachers' knowledge is distilled into the student model. The accuracy of the models on the student test set is used as a metric for comparison. Note that *TEMPDIFF* includes the noise addition step for differential privacy when it infers the public dataset labels for training the student model. As shown in Figure 8, *TEMPDIFF* outperforms the public-only model in classification performance for both datasets. This suggests that knowledge distillation is crucial for classification performance (in *TEMPDIFF*), and equivalent results cannot be obtained by training on a public dataset alone.

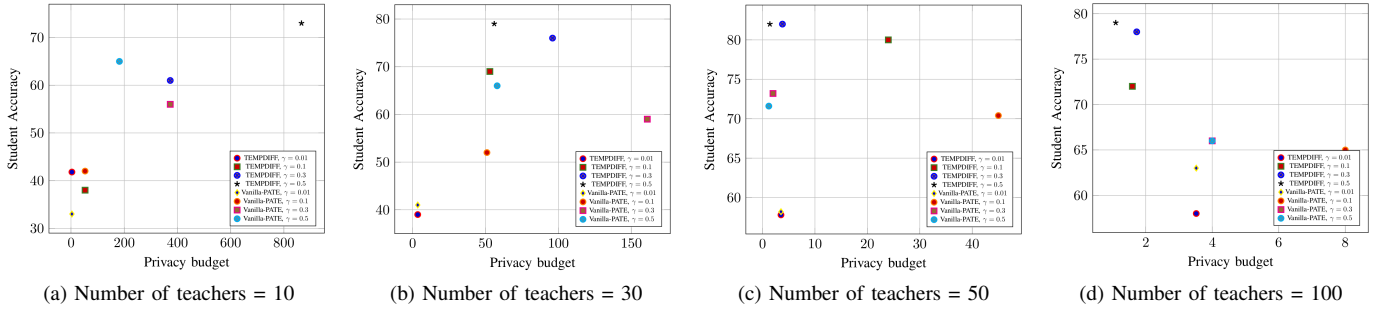


Fig. 6. Privacy budget ( $\epsilon$ ) versus student accuracy (on student model test set) for different  $\gamma$  values of *Vanilla-PATE* and *TEMPDIFF* in WISDM dataset grouped by the number of used teacher models.

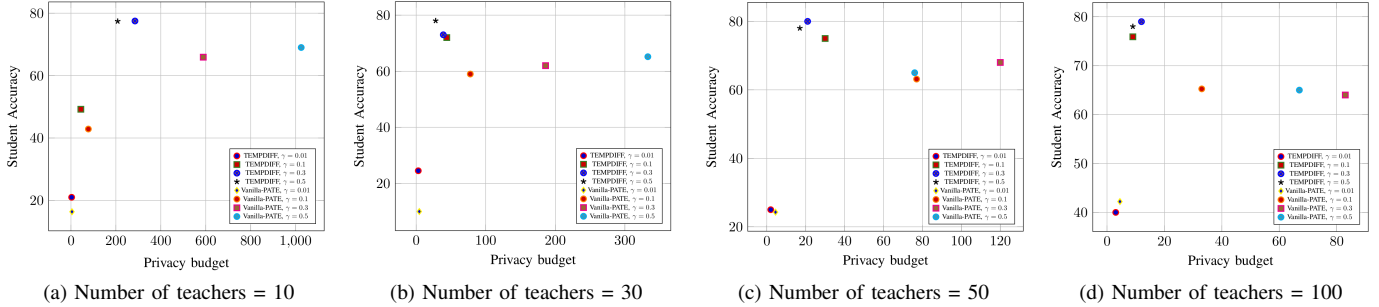


Fig. 7. Privacy budget ( $\epsilon$ ) versus student accuracy (on student model test set) for different  $\gamma$  values of *Vanilla-PATE* and *TEMPDIFF* in PAMAP2 dataset grouped by the number of used teacher models.

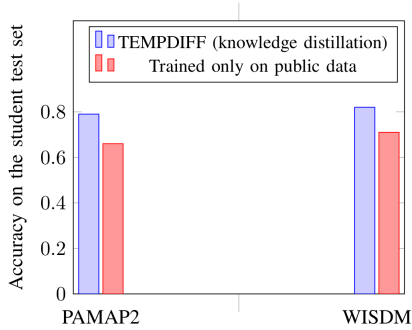


Fig. 8. Effect of knowledge distillation: Accuracy on the student test set for models trained with knowledge distillation (i.e., *TEMPDIFF*) versus models trained only on the public data.

## V. DISCUSSION AND FUTURE WORKS

Our experimental results establish that *TEMPDIFF* framework has the best privacy budget and classification accuracy compared to the state-of-the-art *PATE* baseline. Nevertheless, we observe a gap in the classification performance compared to the baseline. Although it is primarily due to the few examples used to train the student model, the training could be boosted using semi-supervised learning (similar to [22]). In that context, the immediate future direction of this work would be to incorporate self-supervised training in *TEMPDIFF*. Also, we plan to investigate how generative models for student model training as done in [22]. It could be an interesting

future direction for privacy-preserving HAR, as well as because generative models for time-series data need different assumptions compared to vision datasets. While *temporal partitioning* enhances privacy and accuracy, we also observe that it occasionally leads to sub-optimal teacher models. Exploring strategies, such as confidence thresholds [23] or using a mixture-of-experts approach [17], [26], could potentially mitigate these sub-optimal scenarios. Another future direction to explore is creating a collaborative network of privacy-preserving algorithms for HAR tasks. Techniques like federated learning, encryption schemes, differential privacy, etc., could be integrated to form a more robust framework. It would offer different users different privacy preferences in the same framework, e.g., sensitive and non-sensitive users can solve the same downstream task with different privacy preferences.

## VI. CONCLUSION

Human activity recognition (HAR) data originates from a wide variety of users. Data for some users might be sensitive. Thus it is imperative that the privacy of users' data needs to be preserved. It is also important to ensure that the downstream task of human activity recognition does not underperform significantly when privacy is preserved. Hence to satisfy both the performance and privacy requirements, this work proposed a differential privacy framework called *TEMPDIFF* that operates on time-series-based HAR workloads.

*TEMPDIFF* is a differentially private ensembling-based framework that uses the temporality of the sensitive training

data through the *temporal partitioning* algorithm and effectively trains an ensemble of teacher models privately. The aggregated teacher model predictions are differentially private but leak privacy for every prediction. Hence, the teacher ensemble’s knowledge is distilled into a student model using a small public dataset to bind the privacy loss privacy budget. The student model is deployed in public to make differentially private HAR predictions. The framework is evaluated on two public HAR datasets *WISDM* and *PAMAP2*, where it outperforms the state-of-the-art baseline in privacy budget and classification performance. This work would open new avenues of differential privacy research in the HAR and have successful practical, real-world applications.

#### ACKNOWLEDGMENT

We want to thank Vangjush Komini, Lodovico Giaretta, and Thomas Marchioro for the enlightening discussions and for providing feedback for the paper.

#### REFERENCES

- [1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, “Deep learning with differential privacy,” in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016, pp. 308–318.
- [2] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin-Perianu, and P. Havinga, “Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey,” in *23th International conference on architecture of computing systems 2010*. VDE, 2010, pp. 1–10.
- [3] D. Bhattacharya, D. Sharma, W. Kim, M. F. Ijaz, and P. K. Singh, “Ensem-har: An ensemble deep learning model for smartphone sensor-based human activity recognition for measurement of elderly health monitoring,” *Biosensors*, vol. 12, no. 6, p. 393, 2022.
- [4] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, “Differentially private empirical risk minimization,” *Journal of Machine Learning Research*, vol. 12, no. 3, 2011.
- [5] Z. Chen, C. Jiang, and L. Xie, “A novel ensemble elm for human activity recognition using smartphone sensors,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 2691–2699, 2018.
- [6] C. Dwork, “Differential privacy. automata, languages and programming-icalp 2006, Incs 4052,” 2006.
- [7] C. Dwork, F. McSherry, K. Nissim, and A. Smith, “Calibrating noise to sensitivity in private data analysis,” in *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*. Springer, 2006, pp. 265–284.
- [8] C. Dwork, A. Roth *et al.*, “The algorithmic foundations of differential privacy,” *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.
- [9] C. Dwork, G. N. Rothblum, and S. Vadhan, “Boosting and differential privacy,” in *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE, 2010, pp. 51–60.
- [10] A. Garain, R. Dawn, S. Singh, and C. Chowdhury, “Differentially private human activity recognition for smartphone users,” *Multimedia Tools and Applications*, vol. 81, no. 28, pp. 40 827–40 848, 2022.
- [11] Y. Guan and T. Plötz, “Ensembles of deep lstm learners for activity recognition using wearables,” *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, vol. 1, no. 2, pp. 1–28, 2017.
- [12] C. Guo, J. Gardner, Y. You, A. G. Wilson, and K. Weinberger, “Simple black-box adversarial attacks,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 2484–2493.
- [13] J. Hamm, Y. Cao, and M. Belkin, “Learning privately from multiparty data,” in *International Conference on Machine Learning*. PMLR, 2016, pp. 555–563.
- [14] N. Y. Hammerla, S. Halloran, and T. Plötz, “Deep, convolutional, and recurrent models for human activity recognition using wearables,” *arXiv preprint arXiv:1604.08880*, 2016.
- [15] W. Huang, L. Zhang, S. Wang, H. Wu, and A. Song, “Deep ensemble learning for human activity recognition using wearable sensors via filter activation,” *ACM Transactions on Embedded Computing Systems*, vol. 22, no. 1, pp. 1–23, 2022.
- [16] A. Ignatov, “Real-time human activity recognition from accelerometer data using convolutional neural networks,” *Applied Soft Computing*, vol. 62, pp. 915–922, 2018.
- [17] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, “Adaptive mixtures of local experts,” *Neural computation*, vol. 3, no. 1, pp. 79–87, 1991.
- [18] K. A. Kumari, M. Indusha, and D. Dharani, “Enhanced human activity recognition based on activity tracker data using secure homomorphic encryption techniques,” in *2021 2nd International Conference for Emerging Technology (INCET)*. IEEE, 2021, pp. 1–7.
- [19] H. Liu, J. Jia, and N. Z. Gong, “On the intrinsic differential privacy of bagging,” *arXiv preprint arXiv:2008.09845*, 2020.
- [20] L. Lyu and C.-H. Chen, “Differentially private knowledge distillation for mobile analytics,” in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 1809–1812.
- [21] K. Owusu-Agyemeng, Z. Qin, H. Xiong, Y. Liu, T. Zhuang, and Z. Qin, “Msdp: multi-scheme privacy-preserving deep learning via differential privacy,” *Personal and Ubiquitous Computing*, pp. 1–13, 2021.
- [22] N. Papernot, M. Abadi, U. Erlingsson, I. Goodfellow, and K. Talwar, “Semi-supervised knowledge transfer for deep learning from private training data,” *arXiv preprint arXiv:1610.05755*, 2016.
- [23] N. Papernot, S. Song, I. Mironov, A. Raghunathan, K. Talwar, and Ú. Erlingsson, “Scalable private learning with pate,” *arXiv preprint arXiv:1802.08908*, 2018.
- [24] S. Rahman, M. Irfan, M. Raza, K. Moyezullah Ghori, S. Yaqoob, and M. Awais, “Performance analysis of boosting classifiers in recognizing activities of daily living,” *International journal of environmental research and public health*, vol. 17, no. 3, p. 1082, 2020.
- [25] A. Reiss and D. Stricker, “Introducing a new benchmarked dataset for activity monitoring,” in *2012 16th international symposium on wearable computers*. IEEE, 2012, pp. 108–109.
- [26] D. Roy and S. Girdzijauskas, “Mixing temporal experts for human activity recognition,” in *2022 Swedish Artificial Intelligence Society Workshop (SAIS)*. IEEE, 2022, pp. 1–8.
- [27] D. Roy, S. Girdzijauskas, and S. Socolovschi, “Confidence-calibrated human activity recognition,” *Sensors*, vol. 21, no. 19, p. 6566, 2021.
- [28] D. Schuldhuis, *Human activity recognition in daily life and sports using inertial sensors*. FAU University Press, 2019.
- [29] R. Shokri and V. Shmatikov, “Privacy-preserving deep learning,” in *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, 2015, pp. 1310–1321.
- [30] K. Sozinov, V. Vlassov, and S. Girdzijauskas, “Human activity recognition using federated learning,” in *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*. IEEE, 2018, pp. 1103–1111.
- [31] G. M. Weiss, K. Yoneda, and T. Hayajneh, “Smartphone and smartwatch-based biometrics using activities of daily living,” *IEEE Access*, vol. 7, pp. 133 190–133 202, 2019.
- [32] C.-H. H. Yang, J. Qi, S. M. Siniscalchi, and C.-H. Lee, “An ensemble teacher-student learning approach with poisson sub-sampling to differential privacy preserving speech recognition,” in *2022 13th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. IEEE, 2022, pp. 1–5.
- [33] L. Zhang, W. Cui, B. Li, Z. Chen, M. Wu, and T. S. Gee, “Privacy-preserving cross-environment human activity recognition,” *IEEE Transactions on Cybernetics*, 2021.