HHS Public Access

Author manuscript

Conf Rec Asilomar Conf Signals Syst Comput. Author manuscript; available in PMC 2023 August 15.

Published in final edited form as:

Conf Rec Asilomar Conf Signals Syst Comput. 2022 ; 2022: 837–842. doi:10.1109/ieeeconf56349.2022.10052019.

Topological Knowledge Distillation for Wearable Sensor Data

Eun Som Jeon,
Geometric Media Lab, School of Arts, Media and Engineering and School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85281 USA

Hongjun Choi,
Geometric Media Lab, School of Arts, Media and Engineering and School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85281 USA

Ankita Shukla,
Geometric Media Lab, School of Arts, Media and Engineering and School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85281 USA

Yuan Wang,
Department of Epidemiology and Biostatistics, University of South Carolina, Columbia, SC 29208 USA

Matthew P. Buman,
College of Health Solutions, Arizona State University, Phoenix, AZ 85004 USA

Pavan Turaga
Geometric Media Lab, School of Arts, Media and Engineering and School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85281 USA

Abstract

Converting wearable sensor data to actionable health insights has witnessed large interest in recent years. Deep learning methods have been utilized in and have achieved a lot of successes in various applications involving wearables fields. However, wearable sensor data has unique issues related to sensitivity and variability between subjects, and dependency on sampling-rate for analysis. To mitigate these issues, a different type of analysis using topological data analysis has shown promise as well. Topological data analysis (TDA) captures robust features, such as persistence images (PI), in complex data through the persistent homology algorithm, which holds the promise of boosting machine learning performance. However, because of the computational load required by TDA methods for large-scale data, integration and implementation has lagged behind. Further, many applications involving wearables require models to be compact enough to allow deployment on edge-devices. In this context, knowledge distillation (KD) has been widely applied to generate a small model (student model), using a pre-trained high-capacity network (teacher model). In this paper, we propose a new KD strategy using two teacher models – one that uses the raw time-series and another that uses persistence images from the time-series. These two teachers then train a student using KD. In essence, the student learns from heterogeneous teachers providing different knowledge. To consider different properties in features from teachers, we apply an annealing

ejeon6@asu.edu .

strategy and adaptive temperature in KD. Finally, a robust student model is distilled, which utilizes the time series data only. We find that incorporation of persistence features via second teacher leads to significantly improved performance. This approach provides a unique way of fusing deep-learning with topological features to develop effective models.

Keywords

knowledge distillation; topological data analysis; time series data analysis; wearable sensor data

I. Introduction

Wearable sensor data analysis has become increasingly important in various fields such as health and wellness promotion, smart homes, and also intelligent surveillance [1]. Deep learning methods have been utilized to analyze wearable sensor data and have achieved great successes [1]. However, analysis of wearable sensor data suffers from particular issues due to inter and intra-person variability, dependency on temporal segmentation method, and also on the sampling rate of the sensors [2]. To mitigate these issues, invariant features captured by topological data analysis (TDA) have been proposed and have also shown to be effective in stand-alone ways [2].

TDA has been used to characterize the shape of complex data with persistence of connected components and high-dimensional holes decoded by the persistent homology (PH) algorithm [3]. The persistence information can be represented by features such as persistence diagram (PD) and persistence image (PI) [4]. Topological properties are invariant under data deformations such as stretching, bending, and rotation [3], which grants TDA the robustness needed to deal with noisy signal problems and time series data analysis [5]. TDA has been combined with machine learning methods to achieve significant results in time series forecasting [6], stock market analysis [7], [8], disease classification [9], [10], and texture classification [3]. However, TDA requires computational memory and time consumption to extract persistence features from large-scale time series [11], which hinders implementation on small devices with limited computational power.

On the other hand, knowledge distillation (KD) is a promising technique to generate smaller and efficient models (student) by leveraging the learned knowledge from a larger model (teacher) [12]–[14]. During KD, a teacher transfers the knowledge to a student to get the performance as the teacher. It has been applied in wearable sensor data analysis [15] and image classification [11]. Recent studies in KD explore utilizing multiple teachers to generate a better student; to distill a better student, multiple teachers are used with the same data (multi-teacher distillation) [13], [16]–[18]. However, different teachers can generate a ‘knowledge gap’, and there is a chance that data used for training a teacher cannot be utilized to train a student.

In this paper, we propose a new KD strategy to produce an improved student by combining TDA with KD, which alleviates the problem of requiring large computational time and resources for extracting topological features. As described in Figure 1, the proposed method results in a student model that utilizes more informative properties from a diverse set of

teachers – the teachers themselves are learned from the time-series data and their PIs separately. Firstly, to utilize topological features, PIs are obtained from PDs by TDA. We train two separate teachers with the generated PIs and the original time-series data, respectively. Secondly, a student model, that only uses the time-series data as input, is trained by KD from these two teachers. To account for the fact that the teachers have heterogeneous properties, an annealing strategy with adaptive temperature is applied during KD. In this annealing strategy the student model initializes its weights to a model learnt from scratch, instead of random initialization. This helps in reducing the search-space to facilitate fast saturation, and mitigates the knowledge gap between teachers and the student. If not mitigated, this gap can be a hindrance for distillation, since they are trained with different data. Further, to consider different contributions from teachers in the KD loss function, instead of using a fixed temperature value, adaptive temperature values are applied, which are computed from the standard deviation of logits from each network. Finally, a robust and small model is distilled from the proposed method, which uses only the raw time-series data as its input.

The contributions of this paper are as follows:

- We propose a strategy for knowledge distillation transferring topological features of time-series data to a compact student model.
- We develop a technique to reduce the statistical gap in logits between teachers and student for improved knowledge transfer. We present an annealing strategy with adaptive temperature with multiple teachers.
- We show superior empirical results of the proposed method with various teacher-student combinations on wearable sensor data.

II. Background

A. Topological Feature Extraction

TDA has shown robust performance for providing novel insight on the shape of complex data in various fields [4], [5], [7], particularly in machine learning applications [6], [7], [11]. As a key TDA algorithm, persistent homology tracks the changes in topological cavities of different dimensions in data object represented by assortments of points, edges, and triangles through a dynamic thresholding process called filtration [19]. The birth and death of these topological cavities during the filtration are summarized in persistence features, such as persistence diagram (PD) that encodes the birth and death times as x and y coordinates of planar scatter points. Incorporating PDs directly in machine learning models is challenging due to their heterogeneous nature, meaning that the number and locations of the scatter points are not fixed and can vary at the presence of slight perturbations on the underlying data. Ordering the scatter points according to their persistence, or lifetime, through the filtration provides a way of vectorizing the PDs.

Persistence image (PI) is a vector representation of PD motivated as such [4]. To construct the PI, PD is first mapped to a persistence surface $\rho: \mathbb{R} \rightarrow \mathbb{R}^2$, defined by a weighted sum of Gaussian functions centered at the scatter points in the PD. The persistence surface is then

discretized, resulting in a grid. PI is obtained by integrating the persistence surface over the grid and represented with a matrix of pixel values. Higher values of the PI correspond to high-persistence points in the PD. The example of a PD and its corresponding PI are illustrated in Figure 2. However, because of requirements for large memory and time consumption to extract PIs from large-scale signals [11], it is difficult to utilize this method on small devices having limited power and computational resources. In this paper, to solve the problem, we propose a method in knowledge distillation, which trains a smaller model with topological features and generates good performance as a larger model.

B. Knowledge Distillation

Knowledge distillation trains a smaller model by transferring the knowledge from a larger model. KD was first introduced by Buciluǎ *et al.* [20] and developed further by Hinton *et al.* [12]. KD utilizes soft labels from the outputs of a teacher network, which have richer information than just a hard label. Using soft labels helps the student network more easily encode the knowledge from the teacher. For standard KD, the loss function for training a student is:

$$\mathscr{L} = (1-\lambda)\mathscr{L}_C + \lambda\mathscr{L}_K, \tag{1}$$

where $\mathscr{L}_C$ is the standard cross entropy loss, $\mathscr{L}_K$ is KD loss, and λ is a hyperparameter; $0 < \lambda < 1$. The error between the ground-truth label and the output of the softmax layer for a student network is penalized by the cross-entropy loss:

$$\mathscr{L}_C = \mathscr{H}(softmax(l_S), y), \tag{2}$$

where, $\mathscr{H}(\cdot)$ is a cross entropy loss function, l_S is the logits of a student, and y is a ground truth label. The outputs of student and teacher are matched by KL-divergence loss:

$$\mathscr{L}_K = \tau^2 KL(p_T, p_S), \tag{3}$$

where, $p_T = softmax(l_T/T)$ is a softened output of a teacher network, $p_S = softmax(l_S/T)$ is a softened output of a student, and τ is a hyperparameter; $\tau > 1$. Vanilla KD uses a fully trained teacher. Cho *et al.* [14] explored the effects of early stopping for KD (ESKD) to train teacher and student. To obtain the best performance, we adopt early stopping for KD (ESKD) which improves the efficacy of KD [14].

Recently, many approaches for multi-teacher knowledge distillation have been proposed [13], [16]–[18], [21], [22]. The benefit of using multiple teachers is that they can provide different types of useful knowledge to distill a better student. Features from teachers can be used individually or integrated for training a student. In general, since different teachers represent diverse knowledge, richer knowledge can be transferred to a student. On the other hand, there is a possibility that a data point or label used for training a teacher cannot be utilized to train/test a student. Thoker and Gall [23] proposed a method for action recognition in KD to train a student by using paired samples from two modalities. This motivates us to develop and explore similar ideas using topological features involving two teachers. The details of the proposed method are explained in section III.

C. Annealing in KD

Simulated annealing, first introduced by Kirkpatrick *et al.* [24], has been applied to various fields for solving optimization problems [25]. Clark *et al.* [26] presented born-again multitask networks (BAM) utilizing a few single-task teachers to distill a multi-task student network. A dynamic weighted mixture of teacher prediction and ground truth are used for training a student in KD. Early in training, the student model is mostly trained by the teacher, and later, it is mostly trained by hard labels. Annealing KD [27] introduced two stages for KD to solve the capacity gap problem between outputs of teacher and student networks. In the first stage, a temperature parameter decreases as the epoch number grows while the logits of the teacher and student models are matched in a regression task. In stage 2, the student is fine-tuned with the hard labels and standard cross entropy loss. Our annealing strategy in KD uses a different approach compared to prior studies [26], [27]. The proposed method has two teachers that are trained with different data (time-series and persistence-image data) and only one task is learned by the student. Thus, the characteristics of features from teachers are different and their contributions are not the same. To consider their different outputs and contribution, we apply an annealing strategy in the proposed method that reduces the search space and helps for fast saturation, leveraging weights of a network trained from scratch. Also, our method mitigates the gap between outputs of the teachers and student during KD.

III. Proposed Method

The proposed method utilizes two teachers trained from different representations of the same data to distill a better student. First, we extract PIs from time-series data (wearable sensor data). We then train two teachers – one with the raw time-series, and another with the obtained PIs, respectively. Then, to apply annealing strategy, we train a small sized model with learning from scratch, which has the same size as the student. Finally, a student model is trained by an annealing strategy and adaptive temperature for KD, considering differently activated features from teachers.

A. Extracting Persistence Images from Wearable Sensor Data

Leveraging topological features can provide complementary information to improve performance in machine learning. To train a model performing with topological features, we first generate PIs to be used as an input data. We use Scikit-TDA python library [28] and the Ripser package for producing PD. Level-set filtration PDs for time-series signals are computed by the library. Scalar field topology provides a summary for different peaks in the signal. We compute PDs for x, y, z components of the inertial sensor signals in the GENEactiv dataset. We generate PIs, considering its birth-time vs. lifetime information. We set the grid size of the PIs to 64×64 and the parameters for the Gaussian function in PD are the same as reported in Som *et. al.* [11]. The generated PIs resolution is 64×64×3. A model is trained with PIs in supervised mode for classification. Finally, we utilize the topological features as knowledge from this trained teacher to distill a student.

B. Knowledge Distillation with Multiple Teachers

Topological features in and of themselves have been shown to have favorable and also in fusion with other features [3], [7], [10], [11]. However, at test-time, generating PIs from test data adds to computational burden. Our approach results in indirect fusion of topological information via distillation, in a way that at test-time the needed network only operates on raw time-series data, and enjoys low computational load. This approach also avoids the need for concatenation or other hidden layers, for merging the teachers are not necessarily required. This is achieved by knowledge distillation from two teachers simultaneously, trained with time series data and PIs, respectively. To leverage features from two teachers, KL-divergence loss can be written as:

$$\mathscr{L}_{K_m} = \tau^2(\alpha KL(p_{T_1}, p_S) + (1-\alpha)KL(p_{T_2}, p_S)), \tag{4}$$

where α is a parameter to balance the loss values from different teachers, and p_{T_1} and p_{T_2} are softened outputs of teachers trained with time series data and PIs, respectively.

C. Annealing Strategy for Distillation from Multiple Teachers

Since teachers and student are trained with different inputs, features (logits) of teachers have different statistical properties from the one of student. Also, their architectures are different from the student. These differences can create a knowledge gap, which can effect the training of a student [13], [27]. We apply an annealing strategy in KD to reduce the knowledge gap. We train a small model from scratch with the raw time-series data, whose architecture is the same as a student. To start training with KD, student is initialized with the weights of the pre-trained model.

D. Adaptive Temperature for KD

The conventional KD uses fixed temperature τ in the KL-divergence loss term for matching the outputs of student and teacher. However, teachers learned with different data produce differently activated outputs and contributions for each sample to train a student. To consider their different contributions and properties in features, we apply an adaptive temperature using standard deviation for the logit:

$$p' = softmax(l \cdot m/\sigma(l)), \tag{5}$$

where, l is a logit, p' is a softened output, m is a constant for scaling, and $\sigma(\cdot)$ is standard deviation of the input logit. The outputs p' are obtained for teachers and student, respectively. Then, KL-divergence loss is modified as:

$$\mathscr{L}_{k_a} = \tau^2(\alpha KL(p'_{T_1}, p'_S) + (1-\alpha)KL(p'_{T_2}, p'_S)), \tag{6}$$

where p'_{T_1} and p'_{T_2} are outputs from teachers, and p'_S is an output from a student. By this, the outputs from the teachers and student are re-scaled respectively, and mapped to similar ranges while the relationship in-between classes is preserved. Therefore, applying the adaptive temperature also helps to reduce the statistical gap between teacher and student and improve the performance. The softened output of the student p'_S is used with ground

truth for calculating cross-entropy loss $\mathscr{L}_{C_a}$. Finally, the loss function for training the student is:

$$\mathscr{L}_a = (1-\lambda)\mathscr{L}_{C_a} + \lambda\mathscr{L}_{K_a}. \tag{7}$$

IV. Experiments

In this section, we describe dataset and settings for experiments. We evaluate the proposed method with various teacher-student combinations on wearable sensor data.

A. Data Description and Experimental Settings

Data description.—We perform experiments on GENEactiv [29] which is a wearable sensor based activity dataset. We used 14 daily activities, described in detail in a prior study [15]. Each class has over 900 data samples with non-overlapping subjects. The number of subjects for training and testing are over 130 and 43, respectively. We use the data for full non-overlapping window size of 500 time-steps (5 seconds). The number of samples for training and testing are approximately 16k and 6k, respectively.

Experimental settings.—For training a model with time series data, we set the batch size as 64, the total epochs as 200 using SGD with momentum 0.9, a weight decay of 1×10^{-4}, and the initial learning rate lr as 0.05 drops down by 0.2 at 10 epochs and drops down by 0.1 every $[\frac{t}{3}]$ where t is the total number of epochs. For training a model with image data, we set the batch size as 64, the total epochs as 200 using SGD with momentum 0.9, a weight decay of 1×10^{-4}, and the initial learning rate lr as 0.1 drops down by 0.5 at 10 epochs and drops down by 0.2 at 40, 80, 120, and 160 epochs.

In experiments, we use WideResNet (WRN) [30] to construct teacher and student models for evaluating the performance of the proposed method, which is popularly used for KD [14], [15]. The model for training with time-series data consists of 1D convolutional layers. On the other hand, the one with PIs consists of 2D convolutional layers. We determine optimal parameters τ as 4 and λ as 0.7 for KD, following from the previous study [15]. We set a constant m for KL-divergence and cross-entropy as 2 and 1, respectively. α is set as 0.7 to obtain the best result. We run the test 3 times and report with the averaged accuracy and standard deviation for the following experiments.

B. Effect of Teacher Capacity

In this section, we explore the performance of the proposed method with different capacity of teachers. We trained models to be used as teacher models with time-series data and PIs, respectively.

Table I: summarizes the accuracy of various knowledge distillation methods for different combinations of teachers and students. Note, “Time series” and “PImage” denote results of the model trained by KD with Teacher1 trained with time-series data and Teacher2 trained with PIs, respectively. “TS”, “AdTemp.”, and “Ann.” denote using a teacher trained with

time series data, adaptive temperature, and annealing strategy, respectively. The numbers in brackets imply trainable parameters of the model and accuracy, respectively. As shown in the table, our method (TS+PImage with Ann.+AdTemp.) achieves the best performing results in all cases. When teachers from time-series and PIs are used together for distillation, even the basic model performs better than a model trained by learning from scratch and KD with time-series data alone. The results support the idea that annealing strategy and adaptive temperature help improving the performance. Also, using smaller teachers distills a better student, corroborating previous observations [14].

C. Effect of Teacher Combinations

To understand the effect of different teacher architectures, we tested various combinations of two teachers to train a student, considering different channel and depth of networks for WRN. As shown in Table II, in most cases, the proposed method achieves the best results. This further shows that the proposed method helps in improving performance. When WRN16-3 for Teacher1 and WRN28-1 for Teacher2 are used, our method generates a student performing at 71.38% in classification accuracy, that is 3.72% points higher than learning from scratch. Also, the result is better than using teachers with the same depth or width in Table I. In most cases, in Table II, the difference in results from using adaptive temperature is much larger than results in Table I. This is because different teachers generate different features, and create larger knowledge gap between teachers and a student [13], [27]. The results verify that an annealing strategy and adaptive temperature mitigate the knowledge gap and complement to distill a better student.

D. Computational Time

We compare the computational time for various models on the GENEactiv dataset. We implemented the test on a desktop with a 3.50 GHz CPU (Intel® Xeon(R) CPU E5-1650 v3), 48 GB memory, and NVIDIA TITAN Xp (3840 NVIDIA® CUDA® cores and 12 GB memory) graphic card. We tested approximately 6k samples with a batch size of 1. In Table III, the considered accuracies are the best ones from Table I and II. Due to the time needed for generating PIs on CPU, a model learnt from scratch with PIs takes the largest amount of time among the models compared in the table. A WRN16-1 (1D CNNs) student from the proposed method takes the lowest time and shows the best accuracy. The result of CPU underlines the reason why a model compression method such as KD is required for running on edge devices having limited power and computational resources.

V. Conclusion

In this paper, we proposed a strategy in knowledge distillation leveraging topological representations on wearable sensor data, reducing the knowledge gap between the teacher and student by an annealing strategy and adaptive temperature. The proposed method showed more accurate and efficient performance in classification than baselines, and could be significant in various applications on edge devices. In future work, we plan to explore the effect of transferring the topological knowledge from intermediate layers. We also would like to investigate if using more diverse teachers learned with different images (e.g. Gramian Angular Field based images) encoded by time series data would yield a better student.

Acknowledgments

This research was funded by NIH R01GM135927, as part of the Joint DMS/NIGMS Initiative to Support Research at the Interface of the Biological and Mathematical Sciences. Y. Wang is partially supported by the Pilot Project Program of the Big Data Health Science Center at the University of South Carolina.

References

[1]. Nweke HF, Teh YW, Al-Garadi MA, and Alo UR, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," Expert Systems with Applications, vol. 105, pp. 233–261, 2018.

[2]. Seversky LM, Davis S, and Berger M, "On time-series topological data analysis: New data and opportunities," in Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2016, pp. 59–67.

[3]. Edelsbrunner H and Harer JL, Computational topology: an introduction. American Mathematical Society, 2022.

[4]. Adams H, Emerson T, Kirby M, Neville R, Peterson C, Shipman P, Chepushtanova S, Hanson E, Motta F, and Ziegelmeier L, "Persistence images: A stable vector representation of persistent homology," Journal of Machine Learning Research, vol. 18, 2017.

[5]. Wang Y, Behroozmand R, Johnson LP, Bonilha L, and Fridriksson J, "Topological signal processing and inference of event-related potential response," Journal of Neuroscience Methods, vol. 363, p. 109324, 2021. [PubMed: 34428514]

[6]. Zeng S, Graf F, Hofer C, and Kwitt R, "Topological attention for time series forecasting," Advances in Neural Information Processing Systems, vol. 34, pp. 24 871–24 882, 2021.

[7]. Gholizadeh S and Zadrozny W, "A short survey of topological data analysis in time series and systems analysis," arXiv preprint arXiv:1809.10745, 2018.

[8]. Yen PT-W and Cheong SA, "Using topological data analysis (tda) and persistent homology to analyze the stock markets in singapore and taiwan," Frontiers in Physics, p. 20, 2021.

[9]. Pachauri D, Hinrichs C, Chung MK, Johnson SC, and Singh V, "Topology-based kernels with application to inference problems in alzheimer's disease," IEEE transactions on medical imaging, vol. 30, no. 10, pp. 1760–1770, 2011. [PubMed: 21536520]

[10]. Nawar A, Rahman F, Krishnamurthi N, Som A, and Turaga P, "Topological descriptors for parkinson's disease classification and regression analysis," in Proceedings of the Annual International Conference of the IEEE Engineering in Medicine & Biology Society, 2020, pp. 793–797.

[11]. Som A, Choi H, Ramamurthy KN, Buman MP, and Turaga P, "Pinet: A deep learning approach to extract topological persistence images," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2020, pp. 834–835.

[12]. Hinton G, Vinyals O, and Dean J, "Distilling the knowledge in a neural network," in Proceedings of the NeurIPS Deep Learning and Representation Learning Workshop, vol. 2, no. 7, 2015.

[13]. Gou J, Yu B, Maybank SJ, and Tao D, "Knowledge distillation: A survey," International Journal of Computer Vision, vol. 129, no. 6, pp. 1789–1819, 2021.

[14]. Cho JH and Hariharan B, "On the efficacy of knowledge distillation," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 4794–4802.

[15]. Jeon ES, Som A, Shukla A, Hasanaj K, Buman MP, and Turaga P, "Role of data augmentation strategies in knowledge distillation for wearable sensor data," IEEE Internet of Things Journal, vol. 9, no. 14, pp. 12 848–12 860, 2022.

[16]. Reich S, Mueller D, and Andrews N, "Ensemble distillation for structured prediction: Calibrated, accurate, fast-choose three," arXiv preprint arXiv:2010.06721, 2020.

[17]. Liu Y, Zhang W, and Wang J, "Adaptive multi-teacher multi-level knowledge distillation," Neurocomputing, vol. 415, pp. 106–113, 2020.

[18]. Shen C, Wang X, Song J, Sun L, and Song M, "Amalgamating knowledge towards comprehensive classification," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, 2019, pp. 3068–3075.

[19]. Edelsbrunner H, Letscher D, and Zomorodian A, “Topological persistence and simplification,” Discrete Computational Geometry, pp. 511–533, 2002.

[20]. Buciluǎ C, Caruana R, and Niculescu-Mizil A, “Model compression,” in Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD), 2006, pp. 535–541.

[21]. Yuan F, Shou L, Pei J, Lin W, Gong M, Fu Y, and Jiang D, “Reinforced multi-teacher selection for knowledge distillation,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 16, 2021, pp. 14 284–14 291.

[22]. Ni J, Sarbajna R, Liu Y, Ngu AH, and Yan Y, “Cross-modal knowledge distillation for vision-to-sensor action recognition,” in Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 2022, pp. 4448–4452.

[23]. Thoker FM and Gall J, “Cross-modal knowledge distillation for action recognition,” in Proceedings of the IEEE International Conference on Image Processing, 2019, pp. 6–10.

[24]. Kirkpatrick S, Gelatt CD Jr, and Vecchi MP, “Optimization by simulated annealing,” science, vol. 220, no. 4598, pp. 671–680, 1983. [PubMed: 17813860]

[25]. Yang X-S, Nature-inspired optimization algorithms. Academic Press, 2020.

[26]. Clark K, Luong M-T, Khandelwal U, Manning CD, and Le Q, “Bam! born-again multi-task networks for natural language understanding,” in Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2019, pp. 5931–5937.

[27]. Jafari A, Rezagholizadeh M, Sharma P, and Ghodsi A, “Annealing knowledge distillation,” in Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, 2021, pp. 2493–2504.

[28]. Saul N and Tralie C, “Scikit-tda: Topological data analysis for python,” 2019. [Online]. Available: 10.5281/zenodo.2533369

[29]. Wang Q, Lohit S, Toledo MJ, Buman MP, and Turaga P, “A statistical estimation framework for energy expenditure of physical activities from a wrist-worn accelerometer,” in Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2016, pp. 2631–2635.

[30]. Zagoruyko S and Komodakis N, “Wide residual networks,” in Proceedings of the British Machine Vision Conference, 2016.

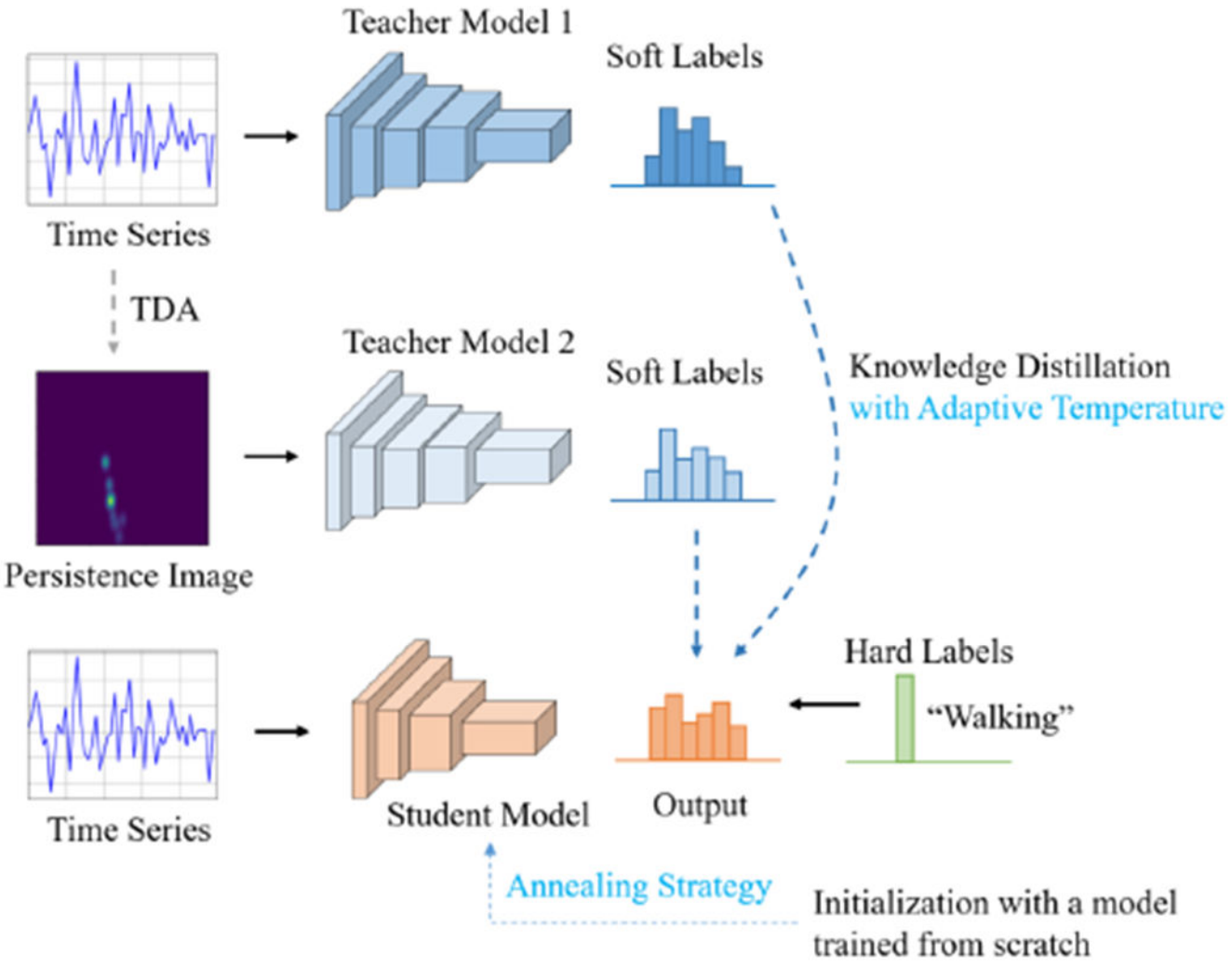

Fig. 1:
An overview of the proposed method. Two teachers learned with different representations of the same raw data are utilized to train a student model.

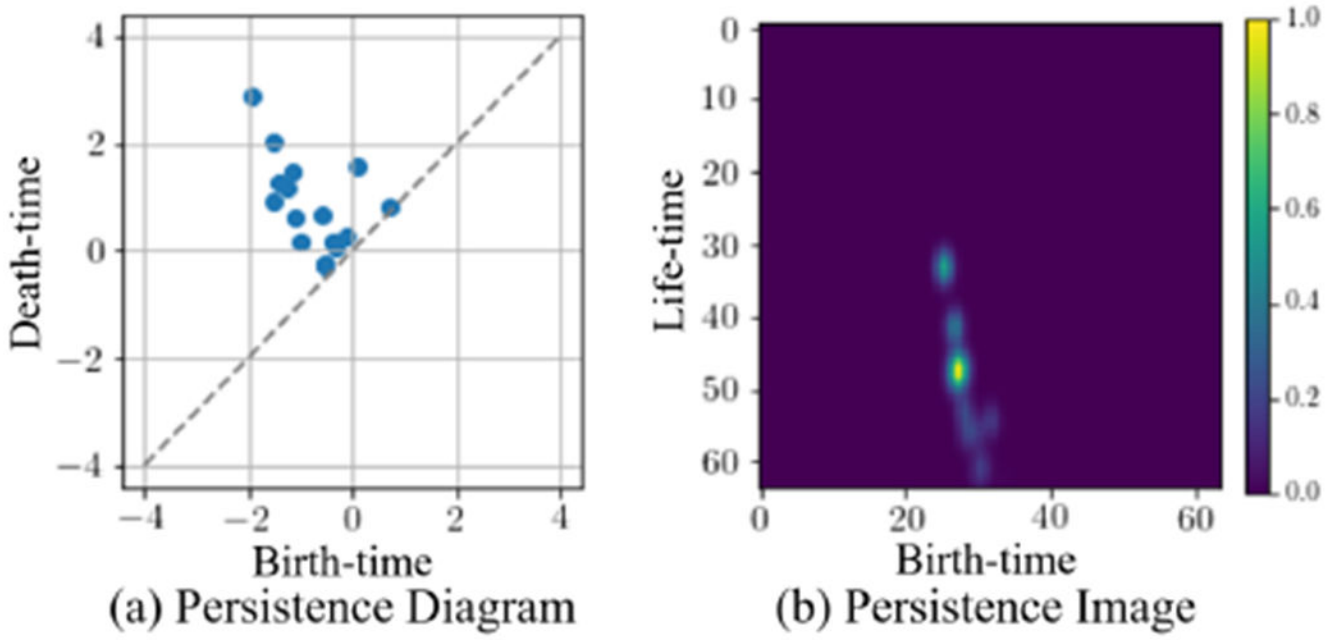

Fig. 2:
PD and its corresponding PI. In PD, higher life-time appears brighter.

TABLE I:

Accuracy (%) with various knowledge distillation methods for different combinations of teachers and students.

Teacher1 (1D CNNs)		WRN16-1 (0.06M, 67.66)	WRN16-3 (0.5M, 68.89)	WRN28-1 (0.1M, 68.63)	WRN28-3 (1.1M, 69.23)
Teacher2 (2D CNNs)		WRN16-1 (0.2M, 58.64)	WRN16-3 (1.6M, 59.80)	WRN28-1 (0.4M, 59.45)	WRN28-3 (3.3M, 59.69)
Student (1D CNNs)		WRN16-1 (0.06M, 67.66±0.45)			
Time series		69.71±0.38	69.50±0.10	67.59±0.36	68.01±0.67
PImage		67.83±0.17	68.76±0.73	68.51±0.01	68.46±0.28
TS+PImage	Base	69.09±0.37	69.24±0.62	69.55±0.41	69.42±0.58
	AdTemp.	69.80±0.68	70.10±0.39	70.01±0.83	69.55±0.51
	Ann.	70.15±0.03	70.71±0.12	70.44±0.10	69.97±0.06
	Ann.+AdTemp.	**70.33**±0.09	**70.86**±0.07	**70.89**±0.12	**70.02**±0.16

TABLE II:

Accuracy (%) with various knowledge distillation methods for different structure of teachers.

Teacher1 (1D CNNs)	Teacher2 (2D CNNs)	Student (1D CNNs)	TS+PImage			
			Base	AdTemp.	Ann.	Ann.+AdTemp.
WRN16-1 (0.06M, 67.66)	WRN28-1 (0.4M, 59.45)	WRN16-1 (0.06M 67.66)	68.71±0.04	69.42±0.56	69.95±0.05	**70.29**±0.11
WRN28-1 (0.1M, 68.63)	WRN16-1 (0.2M, 58.64)		67.89±0.27	69.87±0.55	70.34±0.14	**71.07**±0.05
WRN28-1 (0.1M, 68.63)	WRN28-3 (3.3M, 59.69)		68.25±0.13	69.71±0.22	70.28±0.08	**70.49**±0.17
WRN28-3 (1.1M, 69.23)	WRN28-1 (0.4M, 59.45)	WRN16-1 (0.06M 67.66)	69.09±0.59	69.30±0.29	**69.95**±0.07	69.80±0.09
WRN28-1 (0.1M, 68.63)	WRN16-3 (1.6M, 59.80)		68.04±0.24	69.30±0.13	70.28±0.13	**70.61**±0.15
WRN16-3 (0.5M, 68.89)	WRN28-1 (0.4M, 59.45)	WRN16-1 (0.06M 67.66)	68.87±0.08	70.34±0.54	70.69±0.03	**71.38**±0.05
WRN16-1 (0.06M, 67.66)	WRN28-3 (3.3M, 59.69)		68.15±0.23	68.79±0.25	69.65±0.04	**70.15**±0.11

TABLE III:

Processing time of various models on GENEactiv.

Model	Learning from scratch		KD		Ours (Ann.+AdTemp.)
	TS (1D) WRN28-3	PImage (2D) WRN16-3	TS	PImage	TS+PImage
			WRN16-1 (1D CNNs)		
Accuracy	69.23	59.8	69.71	68.76	**71.38**
GPU (sec)	29.94	356.92 (PIs on CPU)+13.63 (model)	**15.23**		
CPU (sec)	1977.89	356.92 (PIs on CPU)+11191.45 (model)	**16.66**		