Mixing temporal experts for Human Activity Recognition

Debaditya Roy[1], Sarunas Girdzijauskas[2]

***Abstract*— Temporal patterns are encoded within the time-series data, and neural networks, with their unique feature extraction ability, process those patterns to provide a better predictive response. Ensembles of neural networks have proven to be very effective Human Activity Recognition (HAR) tasks with time-series data, e.g., wearable sensors. The combination of predictions coming from the individual models in the ensemble helps boost the overall classification metric through efficient temporal pattern recognition. Currently, the most common strategy for combining the predictions coming from the individual models is *simple averaging*. However, since each ensemble model learns different temporal patterns of the time-series classification problem, a simple averaging strategy is sub-optimal. This sub-optimality is addressed in this paper through a neural network-based adaptive learning framework. The method's core is training a neural gate that ingests the same input time-series data fed to the other temporal models. The goal of the training process is to adaptively learn scaler values against each temporal model by looking at the input data. These scaler values weigh each temporal model while combining the ensemble. The framework obtains superior predictive performance as compared to the standard ensembling techniques. The framework is evaluated on a benchmark HAR dataset called PAMAP2 [3] with two popular state-of-the-art ensemble architectures namely *DTE* [1] and LSTM-ensemble [2]. In both cases, the classification performance of the framework in HAR tasks surpasses the state-of-the-art models.**

I. Introduction

Data from wearable sensors are used in building meaningful applications in ubiquitous and pervasive computing. Human Activity Recognition (HAR) with wearable sensors is one such important application that has gained traction in the recent past. It is a multivariate time-series classification problem where human activities are classified based on the sensor's input data. Since it is a time-series classification problem, neural networks are popular in modelling the task. Our most recent work [1] and another one by Guan et al. [2] achieve state-of-the-art results in this domain through the ensembling of neural network models. In both works, the temporality of the data is exploited during the training of individual models of the ensemble. The models in such ensembles are called *temporal experts* because of their ability to model raw time-series data without any significant pre-processing steps.

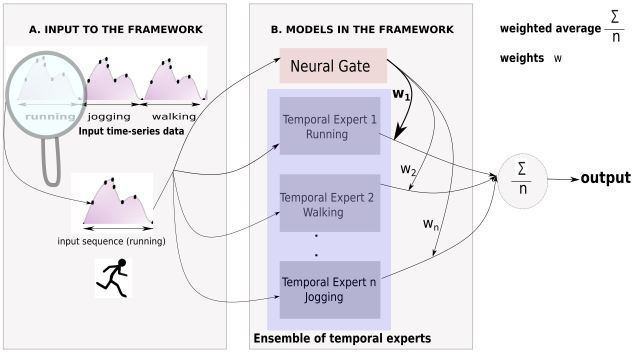


Fig. 1: Mixture of temporal experts framework, that weights the ensembled output using a neural gate through adaptive learning with the input data. Showcased for a running sequence, bold lines indicate higher weight assignment against relevant temporal expert.

For example, in the *Deep time ensembles (DTE)* approach [1], each model in the ensemble become expert in recognizing temporal patterns of different length. These temporal sequences are generated by parsing the raw time-series data (in a sliding window approach) with different window lengths. The hypothesis behind this is based on the fact that different activities in the dataset have different temporal patterns and, hence they might respond to different window lengths. Creating temporal sequences with different window lengths captures these patterns more efficiently. Thus, each model of the ensemble in *DTE* approach is tasked with modelling patterns from different temporal sequences. In *LSTM-ensemble* the models trained using random window-lengths are checkpointed at each epoch and, based on their validation *f1 score*, the best ones are selected for ensembling.

During prediction or test, the models are combined using simple averaging in both cases. While simple averaging of model outputs improved results over the previous state-of-the-art, it does not capture how individual models contributed to the overall prediction. A simple averaging implies equal contribution from all models, but in reality, some models might trigger better predictive responses than others for a particular sequence of input data. For example, a model trained with a temporal sequence of a certain length might capture the running activity more efficiently than others. Thus, in time-series data consisting of different activities, when a temporal sequence of that length for running activity is passed to that model, a better predictive response is

*This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 813162. The content of this paper reflects the views only of their author (s). The European Commission/ Research Executive Agency are not responsible for any use that may be made of the information it contains.

[1]D. Roy is with Department of Electrical Engineering and Computer Science (EECS), Royal Institute of Technology (KTH), Stockholm, Sweden and, with Qamcom Research and Technology, Stockholm.

[2]S. Girdzijauskas is with Department of Electrical Engineering and Computer Science (EECS), Royal Institute of Technology (KTH), Stockholm, Sweden.

expected compared to the other models. While combining the predictions, if no extra importance is given to that particular expert, it dilutes the overall prediction. Hence, although the temporal nature of the data is exploited through ensembling, the naive combination of the temporal experts does not unlock the full potential of the methods. It can be mitigated by assigning higher importance to the experts while combining the ensembles.

In this paper, a framework is proposed that trains a gated neural network responsible for assigning weights to the outputs of the temporal experts, i.e., the pre-trained models of the ensemble. This framework is an improvement over our previous work [1], and it has also shown success in another ensemble-based method in HAR tasks [2]. Figure 1 demonstrates the framework using a sequence from *running* activity as an example. There are n temporal experts, and in this example, it is assumed that they are experts in recognizing certain types of activities such as walking, running, etc. Please note that the temporal experts are not explicitly trained to recognize only a single activity rather input sequences of different lengths [1], but for the sake of explanation, it is assumed in this example. In Figure 1, it is seen that the same input sequence (part A) (running) is fed to n pre-trained temporal experts and, to an untrained neural network called the *neural gate* (part B). The goal of the neural gate is to learn modulating scalar values against each temporal expert for a given input sequence. The outputs of the neural gate are weights $w_1, w_2, ..., w_n$ (Figure 1. These values are used to weigh the output of each of the temporal experts. In this example (Figure 1) model 1 is an expert in recognizing running activity for a given temporal sequence. Hence, when a similar input is given, the neural gate assigns a higher weight w_1 to *temporal expert 1*. It results in a higher contribution from this model during the averaging procedure and hence, a better prediction. The neural gate is a lightweight LSTM architecture that deals effectively with time-series data. The design is made such that the overall training and inference process is fast and straightforward.

The solution is inspired by *mixture-of-experts (MoE)* [9] from the ensemble architecture family. Although *MoE* as a method is used in many domains such as federated learning [8], to the best of our knowledge, in conjunction with raw temporal data in HAR, this is a novel contribution. A discussion on how the proposed method fits in the ensemble landscape and mixture of experts is done in the discussion section.

The framework is tested on a very popular HAR dataset called PAMAP2 [3] with the two ensembles-based modelling approach *DTE* [1] and *LSTM ensembles* [2]. The *f1 score* of the classification task for *LSTM-ensemble* [2] and for *DTE* [1] was 0.85 and 0.89 respectively. The framework improved the *f1 score* to 0.89 in the first case and 0.91 in the second-case. Furthermore, for presenting a robust and unbiased solution, extensive experiments have been performed with statistical tests (Cohen's D test) to establish the significance of the results.

The primary contribution of the work is an adaptive neural network framework that combines temporal experts in an ensemble effectively to improve the predictive performance in Human Activity Recognition tasks.

The paper is organized as follows: Section II discusses the *Related Work*. Following this, an in-depth *Methods* chapter is presented. The *Experiment and Results* are presented in Section IV, followed by a detailed discussion in Section V. Finally, the concluding remarks are presented in the *Conclusion* section.

II. Related Work

Human Activity Recognition with wearable sensor data is a prevalent time-series classification problem explored through different lenses. Popular machine learning algorithms such as Decision Trees, Random Forests, SVMs obtained good results in the task [14], [15], [16], [17]. In recent years deep learning-based methods have become quite popular in modelling this time-series problem [10], [11], [12], [13], [18]. The ability of the deep learning models toward automatic feature extraction leads to the extensive application of these models in Human Activity Recognition. However, some state-of-the-art performances are obtained through the usage of an ensemble of deep learning models instead of using single models [1], [2], [16]. While the performance improvement obtained through the ensemble methods was significant, an in-depth analysis of the method exposes some caveats. For example, most state-of-the-art models use a simple average during model combination, which is a drawback in this scenario. This work addresses the above issue by proposing an adaptive mixture of temporal experts for a better ensemble combination. This method is inspired by the mixture-of-experts framework proposed by Jacobs et al. [9]. In this work, the method is adapted to process a model's expertise in temporal data.

Mixture-of-experts is a popular method that has been used in different domains such as EEG signal classification [19], acoustic data classification [20] stellar data classification [21] and many more. However, all of these methods use statistical inferences over the raw data to address the modelling part of the problem, and none of them work with raw temporal data. Our method addresses models expert in modelling temporal data in the Human Activity Recognition domain. Although mixture-of-experts is a technique that has been used in the domain of Human Activity Recognition before [22], in this case, the raw temporal data is preprocessed to extract statistical features that are fed to individual models in the ensemble setting. In the proposed method, the models employed in the setting learn directly from the temporal data without extra preprocessing.

III. Methods

The paper proposes a neural network-based framework that adaptively scores models of an ensemble for efficient prediction in HAR tasks. It is inspired by the concept of *mixture of experts (MoE)* in an ensemble model [9]. In a *mixture-of-expert (MoE)* ensemble setting, the goal is to divide the existing modelling task into a series of sub-tasks

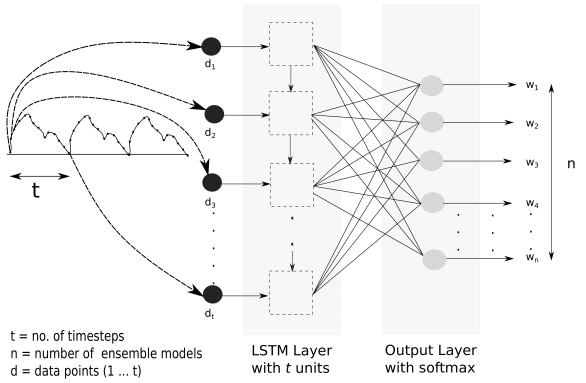


Fig. 2: Architecture of the neural gate that takes in the same time-series input as the temporal experts and outputs weight values against each of the experts. The neural network is composed of a LSTM layer and a fully connected layer.

such that individual models are deployed to solve the sub-tasks. These individual models are experts in solving the sub-tasks assigned to them. An adaptive combination of those experts for a given input can provide an improved generic solver. When sent to such a generic solver, any particular input should be diverted to the relevant experts within that generic solver. Thus, the goal of an MoE system is to solve the overall predictive problem by deploying multiple experts and effectively combining them. The above concept is adopted to improve the predictive performance for time-series workload-based *Human Activity Recognition (HAR)* tasks. In particular, this method extends our previous work *Deep-time-ensembles (DTE)* [1].

The input to *DTE* is a set of window lengths and the raw time-series data originating from wearable devices. Different temporal sequences are extracted from the raw data used to form temporal matrices based on each window length. Each of such temporal matrices is used to train individual models. In *DTE*, individual models of the ensemble are tasked with modelling patterns from different temporal sequences. Thus in this scenario, different models become temporal experts for different sequences. The core idea behind *DTE* is that different temporal experts efficiently model different activities present in the time-series data. The temporal experts are analogous to local experts in the MoE setting. The input time series is divided into temporal sequences based on the highest window length during inference. Moreover, each temporal sequence is fed as a single example. Since each model expects a different size of the temporal sequence, it is up to them to reformulate that input example to the correct size (for more details, we refer to the methods section of our earlier work [1]). Finally, outputs from each model are combined through a simple averaging procedure rendering equal importance to all.

Since each model in the ensemble are experts in specific temporal patterns (that eventually translates into the models being experts at recognizing a particular group of activities), simple averaging is sub-optimal. An unweighted average will assign any extra weight to the softmax response from the experts, thus diminishing any chance of amplifying the correct class prediction originating from the experts. Instead, a weighted average might mitigate the risk by suppressing the undesired classes and amplifying the desired ones. Thus, deploying temporal experts through the MoE framework can mitigate this sub-optimality and improve the overall classification process.

This method aims to improve HAR from time-series data in an ensembling setting leveraging the MoE framework. Doing so would select the correct temporal experts for any given input. As shown in Figure 1, there are n *temporal experts* which are fed with the input time-series data. The time-series data is chopped into temporal sequences, and a prediction is made against each such sequence. For simplicity, it is assumed that each temporal expert specializes in recognizing different activities. Figure 1A depicts such a scenario, where a running sequence is extracted as a single example from input time-series data consisting of multiple activity sequences. This example is fed to the temporal experts and a neural gate. In Figure 1B it is observed that *Temporal Expert 1* is the right choice for the input data, and hence the neural gate assigns higher weight, w_1 to the output of *Temporal Expert 1*. By assigning a higher weight to the correct expert, the output of that expert is amplified in the weighted average resulting in better prediction.

Algorithm 1 Mixture of Experts for Ensembling

1: **Inputs:** n pre-trained models represented as $p_{\theta_i}(y|x)$, the raw time-series data where $i = 1 \dots N$.
2: A untrained lightweight neural network (neural gate) $p_\theta(w_i|x)$ where $i = 1 \dots n$. This neural network outputs a vector w of size n.
3: **for** every point in the raw time-series data. **do**
4: Formulate j temporal sequences t_j
5: **end for**
6: **for** each temporal sequence t_j in t **do**
7: Pass t_j to n pre-trained models and obtain output matrix o_n. Each row of this matrix represents a softmax output of each pre-trained models.
8: Pass t_j to neural gate and get a n sized vector w_n
9: Provide the final output as $\frac{1}{n}\sum_{i=1}^{N} w_n * o_n$
10: Train the neural gate based on this output
11: **end for**

The framework is trained jointly with the pre-trained temporal experts. The objective function is the same as the HAR classification task. The notion is that through this joint training, the neural gate can learn to direct an input temporal sequence to the relevant expert. The algorithm for MOE is presented in Algorithm 1.

Since the crux of the problem is time-series classification and LSTMs [24] are known to process time-series efficiently, the first layer of the neural gate is an LSTM layer with 16

units. The subsequent layer of the neural gate is a dense layer with softmax activation having an equal number of units as the number of temporal experts. Figure 2 depicts the architecture of the neural gate. Temporal sequences are extracted from the raw input time-series data based on a window-size t. Each time a single temporal sequence is fed to the neural gate for prediction. The figure shows a one-dimensional temporal sequence fed into the neural gate by the data points $d_1, d_2, d_3, ...d_t$. The data points pass through the LSTM layer with t units to the output layer. The softmax activation function in the output assigns probabilistic values against each of these n outputs (represented as $w_1, w_2, ..., w_n$ such that the sum of n outputs is 1). These outputs influence the overall classification output during training through the weighted averaging with temporal experts. Thus, the neural gate learns the most effective w_i for different temporal sequences. The importance of each temporal expert is manifested through these values, thus helping in the adaptive model selection process from the ensemble.

A geometrical and mathematical interpretation

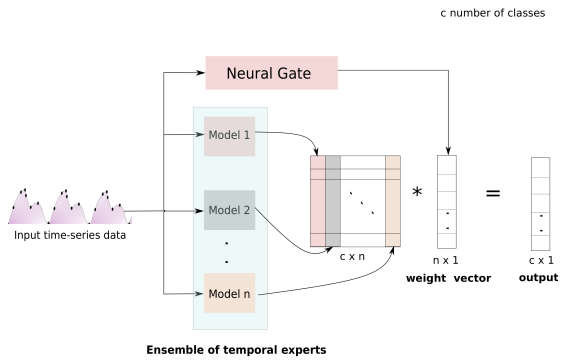


Fig. 3: Mathematical representation of the MoE. This figure shows how the MoE framework is represented as a series of matrix operations.

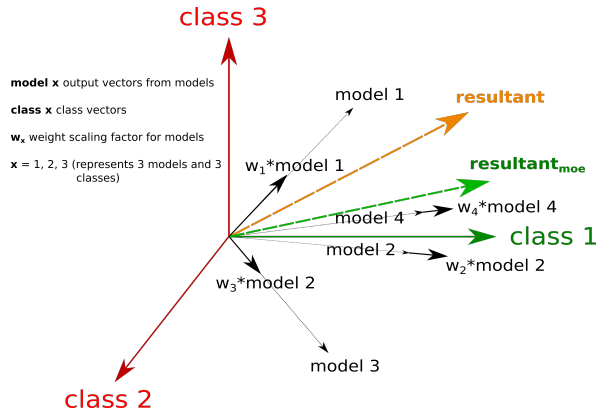


Fig. 4: Geometrical representation of MoE. This figure demonstrates MoE through vector operations in the Euclidian space with a simple example.

Mathematically we can represent the mixture of experts as a series of matrix operations. In Figure 3 the matrix operations for a single example is demonstrated. The outputs of the temporal expert are concatenated into a $c \times n$ matrix, where c represents the number of classes (of the classification task), and n represents the number of temporal experts. Each column of this matrix represents the softmax output for each temporal expert. The output of the neural gate is a $n \times 1$ normalized vector that contains the weighted scores for each temporal expert. The dot product of the matrix and the vector provide us with a $c \times 1$ vector. This vector is the softmax output of the mixture of temporal experts framework for the provided example. It is worth mentioning that the softmax operations at each step can be removed and replaced by a single softmax layer after the dot product.

A geometrical intuition into the mixture-of-experts is shown in Figure 4. In this case, it is assumed that there are three classes namely *class 1*, *class 2*, and *class 3* and there are four temporal experts *model 1*, *model 2*, and, *model 4*. In the diagram, a single example is considered. Since, a single example can belong to only one class at a time the vectors *class 1*, *class 2* and *class 3* are orthogonal. For this example, class 1 is correct, and classes 2 and 3 are incorrect. The dotted vectors represent the predictions provided by each of the temporal experts. For e.g. *model 1* represents the softmax vector output of model 1. The orange *resultant* vector represents the output obtained through simple averaging of each of the softmax vectors. It is seen that the output vectors *model 4* and *model 2* lies close the correct vector *class 1* while the other vectors *model 1* and *model 3* lies closer to the incorrect classes. Doing a simple averaging hence pulls the *resultant* vector away from the directional vector indicating the true class (see Figure 4). Hence, a simple summation would fail to suppress the misleading component and, as a result, diminish the influence of the desired component. As a result, the resultant vector would lie further from the directional vector of the true class when averaged.

Instead, in the MoE setting, a scalar is adaptively learned for each temporal expert. This scaler is used to scale each model vector in its original directions. For example, the w_1 scaler scales the *model 1* vector such that new vector is formed that is given by $w_1 * model\ 1$. The diagram shows that this new vector is of diminished magnitude compared to the original one. New vectors are also calculated using the scalar value for the other models. These new vectors are combined to form the $resultant_{MoE}$ vector that is the final output. The adaptive learning of this scalar value w_1, w_2, w_3 is such that it scales the model output vectors to make the $resultant_{MoE}$ lie as close as possible to *class 1*. Thus, more importance is assigned to the models that are experts in predicting the correct class through this process.

IV. Experiment and Results

This section presents the dataset used for the experiments, the experiments' modelling choices, and the experimental setup.

A. Dataset

PAMAP2 dataset [3] is a popular HAR dataset that consists of 12 activities recorded from nine subjects for ten hours. The primary activities originate from sports (running, walking, etc.) and activities of daily life (ironing, vacuum-cleaning, etc.). The multi-dimensional data is recorded with sensors such as accelerometer, gyroscope, magnetometer, etc., resulting in fifty-two dimensions. Following the same standard established by the previous benchmark papers [1], [2] runs 1 and 2 from subject six are used as testing, and runs 1 and 2 from subject five are used as a validation dataset. It resulted in 83K samples for testing. Using the same evaluation setup allows for a fair comparison with the state-of-the-art methods.

B. Modelling Choices

The architecture of the neural gate is chosen based on the input workload. An LSTM neural network is known to work effectively for time-series data. Since the input workload is time-series data, the neural gate is a single layer LSTM followed by a fully-connected layer with the same number of units as the number of models in the ensemble. Furthermore, to make the neural gate lightweight, the number of chosen LSTM units is 16.

There are two implementation steps involved.

- Train the ensemble of temporal experts.
- Train the neural gate along with the pre-trained temporal experts.

The architectures and hyperparameters for training the temporal experts are adopted from the respective state-of-the-art works (for DTE [1] and for LSTM-ensemble [2]). It is worth mentioning that the *LSTM* architecture from our previous work [1] resulted in a state-of-the-art performance, and hence here it is chosen to train the temporal experts.

The data is also partitioned to fit the two steps. The data is partitioned into training, validation and testing. The ensemble models are trained using 80 percent of the training data, and the rest 20 percent is used to train the neural gate. The final performance reporting is done on the same testing data partition as the benchmark.

The neural-gate is trained using the Adam optimizer [23] with a learning rate of 0.001 for ten epochs. Since the only weights modified through the backpropagation belong to the neural gate, the training process is fast and short.

C. Experimental setup

The approach is tested on the popular HAR dataset of PAMAP2 [3], and on two ensembling approaches namely, DTE [1] and LSTM-ensembles [2]. In both cases, individual models are trained based on the proposed approaches. These pre-trained models are then trained with the gated neural network. The classification performance of the proposed method towards Human Activity Recognition is measured using the *macro f1-score* and *accuracy* metrics. The experiments were repeated 20 times, and the mean performances were reported for the benchmark and the proposed approach. Furthermore, the result of the compared benchmark and the proposed framework are fed into a two-tailed Kolmogorov-Smirnov test [25] to check whether the distributions are the same or not. If they indeed are different, the same data is fed into a Cohen's d test to compute the effect size [26]. The Cohen's d test results help understand if the classification result of the proposed framework is different enough to be statistically significant compared to the competitors. The whole evaluation setup is directed towards fairness and robustness.

D. Results

TABLE I: Comparison of accuracy results between *DTE*, *LSTM-ensembles* and, *our method*

Method	F1 score	Accuracy
LSTM-ensembles	0.85 ± 0.01	0.85 ± 0.009
DTE	0.88 ± 0.003	0.89 ± 0.003
Our Method	$\mathbf{0.9 \pm 0.003}$	$\mathbf{0.91 \pm 0.004}$

TABLE II: Cohen's D-test comparison of *DTE* and *LSTM-ensembles* with *our method*

Method	Cohen's D
LSTM-ensembles	5.11
DTE	1.5

TABLE III: Comparison between *LSTM-ensembles* and, *our method* for the number of models required

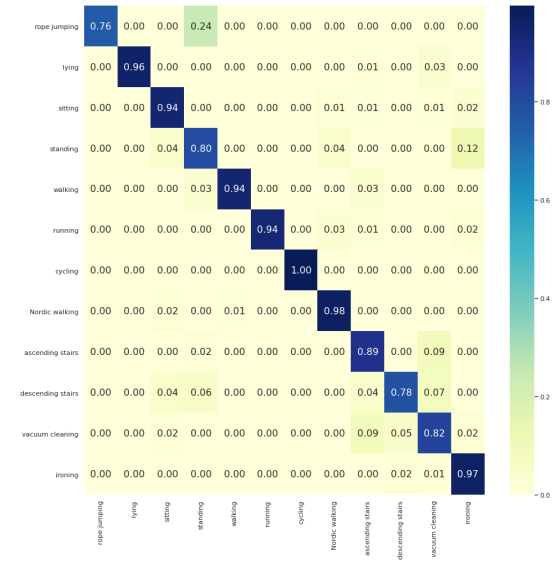
Method	Number of ensemble models	F1 score
LSTM-ensembles	20	0.86 ± 0.008
Our Method	**5**	$\mathbf{0.89 \pm 0.003}$

The results presented in Table I shows the improvement of *f1 score* and *accuracy* achieved by the proposed method. In particular, the proposed MoE framework increases the accuracy and f1-score by 6% against [2] and 2% against our previous work [1]. Table II shows the statistical significance of our tests. For each comparison, we report the Cohen's D test values. As both the values are significantly bigger than 0.4, we conclude that the difference achieved in *f1 score* is significant enough to be reported as an improvement over the state-of-the-art.

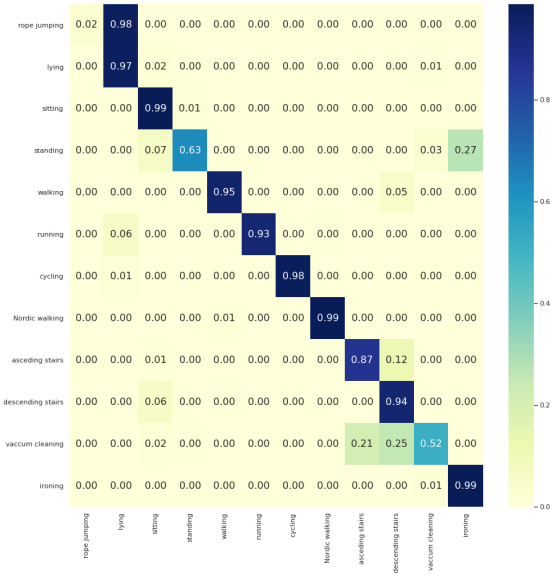
Table III presents an experiment that shows that when compared with *LSTM ensembles* [2] our method requires much lesser number of models in the ensemble to achieve an equivalent result. The confusion matrices of the results when using MoE over both the benchmark methods are shown in Figure 5.

V. Discussion

The adaptive combination of temporal experts in the Human Activity Recognition problem is addressed in this work. Although originally intended to extend our previous work [1], it is shown to be effective on other ensemble architectures for HAR tasks as well. This chapter discusses the results and insights of the proposed method.



(a) Confusion Matrix: MoE on DTE



(b) Confusion Matrix: MoE on LSTM-ensemble

Fig. 5: Confusion matrices of results

A. Extension of MOE on DTE

Deep time ensembles (DTE) [1] tried to model the HAR problem by exploiting temporal patterns that exist in the time-series data. In particular, in this method, each of the models in the ensemble is an expert in recognizing different temporal patterns extracted from the raw time-series data. Thus, in a mixture-of-experts framework, the expertise of each model lies in temporal pattern recognition. Through the neural gate, expertise is propagated during the model combination stage of ensembling which is evident from the results in Table I. We see that through a weighted averaging, we get an improvement in both accuracy and the f1-score as opposed to a simple averaging of standard *DTE*. Thus, through a simple mechanism, this method can improve *DTE*.

B. Extension of MOE on LSTM-ensemble

LSTM-ensemble [2] being a standard ensemble based benchmark in the field of HAR is a fit candidate for extension with mixture-of-experts. However, in this particular method, there is no systematic way to attribute expertise to individual models of the ensemble. Each ensemble model is a by-product of progressive checkpointing of every epoch and randomized window-length selection. This makes it intuitively harder to assign sub-problem expertise to each model. During combination, based on the best validation accuracy of each model, results from the top n models are averaged. While in *DTE* the temporal experts can be related with a particular window length, there is no such scope of doing that in this case. However, randomized window lengths fed to the experts can also be lightly associated with the temporal expertise of each model. As observed in Table I here also, the proposed method outperforms the existing benchmark. Furthermore, it is also seen that the proposed method needs a lesser number of models to outperform the benchmark (Table III). It further indicates that a specialized selection through the proposed framework helps better and targeted model selection.

C. Where does it fit in the ensemble landscape?

Ensemble-based methods are used in machine learning extensively [4]. The commonly used ensemble methods are *bagging, boosting, mixture-of-experts* [7] etc.

In a *bagging* strategy, diverse models are created through bootstrapping of different subsets of training data. Whereas in *boosting* the subsequent models are boosted based on the performance of the present model. In particular, the data is chosen for subsequent models geared towards this strategy. The *mixture-of-experts* is a meta-learning method where the contribution of each model in the ensemble is measured via a gating network. An investigation of the proposed models with respect to the above three ensembling techniques is explored in this subsection.

In this paper, two types of ensembling techniques are discussed. This section evaluates the relationship between the standard ensembling procedures and the two methods.

The *first modelling strategy* as discussed in the *LSTM-ensemble* paper [2] involves multiple models derived from each epoch of training and their subsequent combination through an averaging procedure. For the combination, n best models are selected out of all those saved for every training epoch. This method can be seen as combining multiple weak learners. However, since the dataset that is used for training the models is precisely the same, it is not exactly a standard *bagging* method. The only similarity it has with bagging is that the *neural network* parameters; hence the *models* are the same in this case. It can be categorized partially as a *boosting* method because, for each epoch, the models are assumed to be better than the previous one till the learning process converge. As opposed to standard *boosting* in this case, there

is no resampling of data. The main advantage of this method is that it is extremely fast as none of the models are fully-trained models, but they represent a progressive snapshot of the training process.

The *second modelling* strategy is *Deep time-ensembles* as discussed in [1]. In this ensembling method, the same input time-series data is re-represented differently for each model in the ensemble. This re-representation allows extraction of different types of temporal patterns, which results in improved predictive performance. This method can be partially linked with *bagging* because the same type of model is trained with a different representation of the same data. The partial relationship is attributed to the fact that different subsets of data are used as input for different models; however, a different representation is used in this case. The models, in this case, are also combined with a simple algebraic average.

In both these cases, a combination of the models through a *neural gate* can be seen as the *mixture of experts* (moe) ensemble modelling. In a mixture of experts through a divide and conquer approach, the problem space is divided into multiple subspaces, and a modelling strategy is adopted to solve each of those subspaces. In this problem setup, the problem subspaces are separated by placing temporal assumptions, and each model in the ensemble is tasked to solve problems with different temporality. In the combination step, through some adaptive methods, the models are combined. Usually *gating network* is trained through expectation-maximization or EM algorithm [6] serves as the adaptive combiner. In our case, a *neural-gate* trained with backpropagation serves as the adaptive ensemble combiner.

VI. Conclusion

This paper introduced a simple yet elegant solution to combine temporal experts in an ensemble for better predictions in HAR tasks. This framework aims to adaptively learn weight values for each temporal expert against different types of inputs. The solution entails training a neural gate and pre-trained ensemble models together for the same input data with the classification objective of HAR. The training of this network is very fast because the only trainable component of this network is the neural gate composed of a lightweight LSTM network. Thus, the primary advantage of this approach is that it provides better predictions at the expense of a meager computational cost.

The nature of the problem this method tackles is that of time-series classification. In particular, it is used in the domain of Human Activity Recognition from wearable sensor data. The method arises from the family of *mixture-of-experts* in ensembles and exploits the temporal expertise of the HAR models to combine them using a *neural gate*. To the best of our knowledge, no previous works have used the temporal *mixture-of-expert* variant in HAR tasks. We have chosen a standard benchmark HAR dataset PAMAP2 [3] that consists of data from multiple wearable sensors and twelve activities. The method is applied with state-of-the-art ensemble-based approaches to this dataset and has improved classification results. Even though the proposed method can improve predictions in HAR tasks, it also opens up new avenues to explore temporal tasks in other domains such as financial forecasts, video-RGB data-based time-series tasks, etc.

ACKNOWLEDGMENT

We would like to extend our thanks to Vangjush Komini (from KTH Royal Institute of Technology, Stockholm and Qamcom Research and Technology, Stockholm), who reviewed the paper and helped formalize the mathematical and geoetrical intution of the mixture-of-experts framework in the context of *DTE* (Presented in Methods section).

References

[1] Roy, D., Girdzijauskas, S. and Socolovschi, S., 2021. Confidence-calibrated human activity recognition. Sensors, 21(19), p.6566.

[2] Guan, Y. and Plötz, T., 2017. Ensembles of deep lstm learners for activity recognition using wearables. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 1(2), pp.1-28.

[3] Reiss, A. and Stricker, D., 2012, June. Introducing a new benchmarked dataset for activity monitoring. In 2012 16th international symposium on wearable computers (pp. 108-109). IEEE.

[4] Valentini, G. and Masulli, F., 2002, May. Ensembles of learning machines. In Italian workshop on neural nets (pp. 3-20). Springer, Berlin, Heidelberg.

[5] Robi Polikar (2009) Ensemble learning. Scholarpedia, 4(1):2776.

[6] Moon, T.K., 1996. The expectation-maximization algorithm. IEEE Signal processing magazine, 13(6), pp.47-60.

[7] Bühlmann, P., 2012. Bagging, boosting and ensemble methods. In Handbook of computational statistics (pp. 985-1022). Springer, Berlin, Heidelberg.

[8] Zec, E.L., Mogren, O., Martinsson, J., Sütfeld, L.R. and Gillblad, D., 2020. Specialized federated learning using a mixture of experts. arXiv preprint arXiv:2010.02056.

[9] Jacobs, R.A., Jordan, M.I., Nowlan, S.J. and Hinton, G.E., 1991. Adaptive mixtures of local experts. Neural computation, 3(1), pp.79-87.

[10] Ordóñez, F.J. and Roggen, D., 2016. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. Sensors, 16(1), p.115.

[11] Ignatov, A., 2018. Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. Applied Soft Computing, 62, pp.915-922.

[12] Jiang, W. and Yin, Z., 2015, October. Human activity recognition using wearable sensors by deep convolutional neural networks. In Proceedings of the 23rd ACM international conference on Multimedia (pp. 1307-1310).

[13] Murad, A. and Pyun, J.Y., 2017. Deep recurrent neural networks for human activity recognition. Sensors, 17(11), p.2556.

[14] Bao, L. and Intille, S.S., 2004, April. Activity recognition from user-annotated acceleration data. In International conference on pervasive computing (pp. 1-17). Springer, Berlin, Heidelberg.

[15] Casale, P., Pujol, O. and Radeva, P., 2011, June. Human activity recognition from accelerometer data using a wearable device. In Iberian conference on pattern recognition and image analysis (pp. 289-296). Springer, Berlin, Heidelberg.

[16] Feng, Z., Mo, L. and Li, M., 2015, August. A Random Forest-based ensemble method for activity recognition. In 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 5074-5077). IEEE.

[17] He, Z. and Jin, L., 2009, October. Activity recognition from acceleration data based on discrete consine transform and SVM. In 2009 IEEE International Conference on Systems, Man and Cybernetics (pp. 5041-5044). IEEE.

[18] Yao, S., Hu, S., Zhao, Y., Zhang, A. and Abdelzaher, T., 2017, April. Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In Proceedings of the 26th international conference on world wide web (pp. 351-360).

[19] Guler, I., Ubeyli, E.D. and Guler, N.F., 2006, January. A mixture of experts network structure for EEG signals classification. In 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference (pp. 2707-2710). IEEE.
[20] Nguyen, T. and Pernkopf, F., 2019, July. Acoustic scene classification with mismatched recording devices using mixture of experts layer. In 2019 IEEE International Conference on Multimedia and Expo (ICME) (pp. 1666-1671). IEEE.
[21] Jiang, Y. and Guo, P., 2005, May. Mixture of experts for stellar data classification. In International Symposium on Neural Networks (pp. 310-315). Springer, Berlin, Heidelberg.
[22] Lee, Y.S. and Cho, S.B., 2014. Activity recognition with android phone using mixture-of-experts co-trained with labeled and unlabeled data. Neurocomputing, 126, pp.106-115.
[23] Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
[24] Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.
[25] Massey Jr, F.J., 1951. The Kolmogorov-Smirnov test for goodness of fit. Journal of the American statistical Association, 46(253), pp.68-78
[26] Becker, L.A., 2000. Effect size (ES).