Adaptive Early Classification of Time Series Using Deep Learning

Anshul Sharma[1(✉)], Saurabh Kumar Singh[2], Abhinav Kumar[3], Amit Kumar Singh[4], and Sanjay Kumar Singh[3]

[1] The LNM Institute of Information Technology, Jaipur, Rajasthan, India
anshul@lnmiit.ac.in
[2] Tezpur University, Tezpur, Assam, India
[3] Indian, Institute of Technology (BHU) Varanasi, Varanasi, India
{abhinav.rs.cse17,sks.cse}@iitbhu.ac.in
[4] National Institute of Technology Patna, Patna, India
amit.singh@nitp.ac.in

Abstract. Early Classification of Time Series (ECTS) is a process of predicting the class label of time series at the earliest without observing the complete sequence. Time Series data is a collection of data points over time, and a decision has been made based on a complete sequence. However, early decision based on partial information is beneficial in time-sensitive applications. ECTS is an emerging research area with multiple applications in various domains such as health and disease prediction in medicine, Quality and Process Monitoring in Industry, Drought and Crop monitoring in agriculture. In this paper, we propose an adaptive early classification model composed of two components. The first component is the base classifier, which has been designed as a hybrid model of Convolutional Neural Network and Recurrent Neural Network. The Second component is the decision policy designed for adaptive halting capabilities, which has been defined as a reinforcement learning agent to determine when to stop and make a prediction. We evaluated our model on publicly available different kinds of time-series datasets. The proposed method outperformed the state-of-the-art in terms of both accuracy and earliness.

Keywords: Time Series · Early Classification · Deep Learning · RNN · CNN · Reinforcement Learning

1 Introduction

Classification of time series is a prominent problem in temporal data analysis. Time series is the sequence of observations collected/measured over time. Many time series classification algorithms have emerged and offer cutting-edge solutions [1]. The primary aim of time series classification is to predict the class level of given time series. In the traditional classification approach, time series are classified when the complete sequence becomes available. However, early decisions

M. Tanveer et al. (Eds.): ICONIP 2022, LNCS 13625, pp. 533–542, 2023.
https://doi.org/10.1007/978-3-031-30111-7_45

based on partial information are highly beneficial in time-sensitive applications, e.g., detecting high-risk events like a market crisis or earthquakes in advance. Applications of early classification are found in many areas, including agriculture [2], industries [3,4], medicine [5], Security [6] and transportation [7].

ECTS aims to classify the time series as early as possible based on incomplete sequences while ensuring an acceptable level of accuracy [8,9]. The problem of ECTS has two conflicting objectives accuracy and earliness. If we have more observations, we are more reliable about the prediction and vice-versa. Thus, the earliness of prediction is often inversely related o the accuracy, and balancing the trade-off between these two makes this problem challenging.

Rodríguez Diez and Alonso Gonzalez [10] introduced the problem of ECTS first time, and since then, the problem of ECTS has gained popularity among the research community [2,9,11–14]. They used relative and region-based predicates on segmented intervals from the time series. Predicates such as increments, stays, and decrements were relative predicates, and always, sometimes and true-percentage were region-based predicates. They constructed classifiers using predicated features, each of them containing only one predicate. To overcome the problem of incomplete data, they used Adaboost to make the base classifiers, which could make predictions on incomplete data. The significant contribution of this work was to identify the importance of the ECTS. Xing et al. [11] formally defined the ECTS and developed 1-NN based early classifier that examines the closest neighbour relationship in the training set. They also pointed out that the most critical aspect of early classification is to make the balance between earliness and accuracy. In another study, Xing et al. [15] presented a shapelet-based method named EDSC. Shapelets are the subsequence of time series defined as class representatives. The proposed method identified the shapelets with unique features in time series and learned to discriminate between classes as early as possible. In Ref. [16], the authors presented a method named Reliable Early Classification (RelClass) which utilized quadratic discriminant analysis (QDA) for early classification. The method used a user-defined threshold to determine a reliability score which is estimated as the probability that the true class level of complete time series and predicted class of truncated will be the same or higher at each timestep. In [8], the authors proposed an early classification approach based on differentiating classes over time, named ECDIRE. They defined the reliability threshold as a probabilistic difference between the two most probable classes and used it as a decision criterion for making an early prediction.

In recent years neural networks have been extensively used to classify time series and have shown promising results [9,17]. Ref. [9] presented an early classification model that uses LSTM as a classifier and a reinforcement learning agent for decision policy. This model focuses on optimizing both objectives simultaneously, which are accuracy and earliness.

1.1 Motivation and Contributions

Time-sensitive applications are major motivations for early classification. These applications can be found in every field, e.g., medical diagnosis of diseases with distinct symptoms at different timestamps, human activity classification, industrial process monitoring, and electricity usage monitoring. Literature indicates that there are many applications of ECTS in data mining and machine learning [2–4,9,14].

A recent application of early classification could be the detection of COVID-19 infected persons [18] during the pandemic, which is very crucial to stop the spread of the Virus. These applications directly impact the lives of the mass population and possibly save lives. Here we propose an adaptive early classification approach for sequence data. Contributions to our work are:

– We design a novel ECTS method that handles the Early classification problem in two parts. The first part extracts information from the partially observed time series using a base classifier, and the second part makes use of a Reinforcement Learning (RL) Agent to determine whether enough information is captured to make a prediction or not.
– The base classifier has been designed as a hybrid model of RNN and CNN that can capture temporal as well as spatial dependency from incomplete time series.
– The proposed solution utilized the specialty of RNN, CNN, and RL for adaptive early classification by optimizing both accuracy and earliness.
– We evaluated our proposed RCRL (RNN, CNN and Reinforcement Learning) model using seven datasets taken from the UCR Time series classification archive [19]. Our method provided a good balance between accuracy and earliness and outperformed state-of-the-art alternatives in terms of these objectives.

2 Methodology

We propose an early classification model by utilizing the characteristics of RNN, CNN, and RL, named RCRL. The proposed model architecture is depicted in Fig. 1. The RCRL model is logically composed of two components: a base classifier and a decision policy. The proposed RCRL model discriminates between classes based on the extracted features from the RNN and CNN blocks; these features are then fed to a reinforcement learning agent, which learns an early classification policy based on the observed information. The proposed model captures temporal as well as spatial information from the time series. A recurrent neural network helps to capture temporal information, and CNN helps to capture spatial information. The proposed model combined these networks as a hybrid classifier similar to MLSTM-FCN [17]. Further, this base classifier is combined with a reinforcement learning agent to make an adaptive early-classifier [9].

2.1 RCRL Model

We propose an RCRL model that uses an RNN block and a CNN block to extract features from the input data at each timestamp. The extracted features are then passed through an RL Agent, which decides whether or not to halt and predict a

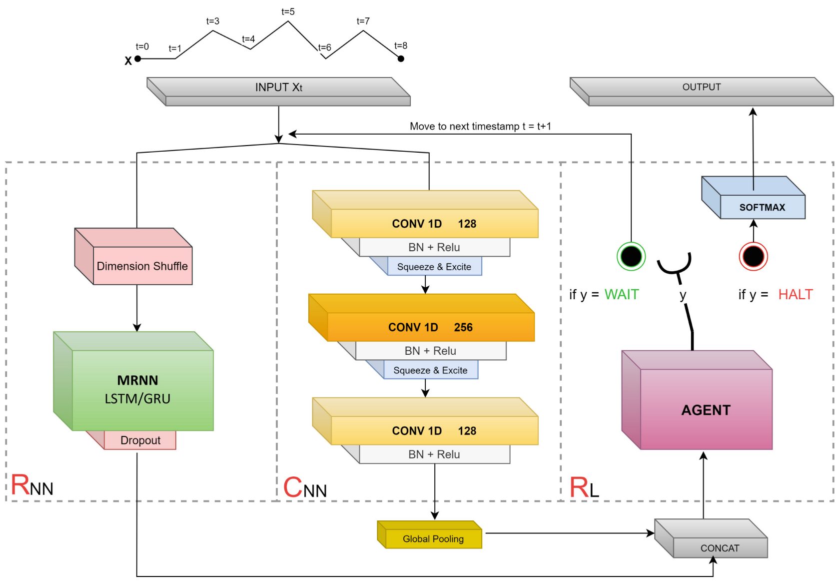

Fig. 1. RCRL-model

class label based on these input features. The agent solves a Partially-Observable Markov Decision Process (POMDP) in which the features extracted by the RNN-CNN block are received at each timestamp. Depending on the agent's learned policy, action is sampled, and based on the sampled action's quality, a reward is observed. The agent's job is to maximize its reward based on accuracy and earliness of prediction. The idea of decision policy is adopted from [9]. However, It has been learned with different kinds of features.

The agent's action a controls the halting of classifier proceedings, the agent can takes two actions $a = \{0, 1\}$, where $a = 1$ dictates the classifier to halt and produce a prediction and $a = 0$ dictates to continue observing more timestamps by classifier. We use ϵ-greedy action to minimize heavy exploitation by the agent. The value of ϵ decreases exponentially from 1 to 0. During training, random action is sampled with probability ϵ as shown in Eq. (1).

$$a_t = \begin{cases} a_t, & \text{with probability } 1 - \epsilon \\ \text{random action}, & \text{with probability } \epsilon \end{cases} \tag{1}$$

Current state H_t contains information about the extracted features by the classifier up to t timestamps. An action is sampled using the learned policy by the agent; the policy is learned using function approximation which maps state H_t to the probability of halting P_t.

The accuracy and earliness of the classification work against each other, while focusing too much on earliness, accuracy gets severely compromised. To address this problem of accuracy-earliness trade-off a hyperparameter λ has been introduced, which controls the earliness of the model by penalizing for late halting, this penalty addition to the loss function along with classification loss motivates the agent to classify early and also maintain the classification accuracy.

2.2 Network Architecture

The CNN block is worked as a feature extractor, it contains 3 temporal convolutional blocks, each block has a convolutional layer of kernel sizes 7, 5, and 3 and with several filters (128, 256, and 128). The next layer after each convolutional layer is Batch Normalization layer(momentum is 0.99 and epsilon is 0.001) followed by a ReLU activation function. At the end of the first two convolutional block a squeeze-and-excite [20] is added with a reduction ratio of 16. At the end of the third convolutional layer, a global pooling layer is added.

In addition to the CNN block, an RNN (LSTM [21] or GRU [22]) block is also introduced as a feature extractor. The RNN block contains a dimension shuffle layer, followed by an RNN layer and a dropout layer. The features extracted by these blocks are concatenated and then passed through an RL block, the RL agent learns a stochastic policy using a 1 layer fully-connected neural network to decide the halting timestamp at which the softmax function will be applied to produce the final class predictions.

2.3 Training

We trained the RNN and CNN block together first and then we used the trained model's parameters to train the reinforcement learning agent. The agent is trained separately on the extracted features from the trained RNN and CNN block, if the learned halting policy of the agent decides to halt at a certain timestamp the extracted features are passed through a softmax layer to produce the final prediction.

The model aims to maximize accuracy while training the classifiers, which are the RNN and CNN blocks. After that, we fixed the parameters of the trained model and introduced the RL agent into the model. While teaching the RL agent, its aim is to maximize the observed reward for classification. The reward function of the agent has a term introduced into it to penalize for earliness, the weightage of the penalty is controlled by hyperparameter λ. The higher value of λ yields a trained model with a high priority on earliness and low priority on the accuracy, while a lower value of λ yields a trained model with a high priority on accuracy and low priority on earliness.

3 Experimental Evaluations

The proposed RCRL has been evaluated on 7 datasets from the UCR archive [19], these datasets have also been used in the evaluation of prior related works on ECTS. We compared RCRL with state-of-the-art methods including, TEASER [23], ECTS [11], RelClass [24], EDSC [15] and ECDIRE [8] as shown in Table 1.

Table 1. Accuracy Comparison

Dataset	ACCURACY(%)					
	RCRL	TEASER	ECDIRE	RelClass	ECTS	EDSC
MoteStrain	65	89	80	58	88	78
ItalyPowerDemand	67	67	93	85	94	82
SonyAIBORobotSurface2	78	78	74	88	85	81
TwoLeadECG	80	77	81	72	73	88
SonyAIBORobotSurface1	85	85	83	79	69	80
ECG200	88	86	91	89	89	85
GunPoint	87	84	87	91	87	94
Dataset	EARLINESS(%)					
	RCRL	TEASER	ECDIRE	RelClass	ECTS	EDSC
MoteStrain	8.7	21	12.1	90.94	79.06	38.08
ItalyPowerDemand	19.63	22	70.16	35.92	79.33	67.08
SonyAIBORobotSurface2	15.9	21	17.66	70.86	54.54	35.51
TwoLeadECG	21	21	69.38	83.63	64.43	46.85
SonyAIBORobotSurface1	8.7	28	62.26	57.7	68.49	47.03
ECG200	9.7	20	90.1	68.81	60.11	23.24
GunPoint	16.8	23	32.37	71.33	46.92	45.58

The results of these methods have been taken from the source[1] for comparison, where these results have been recomputed by running the source codes of these papers.

3.1 Evaluation Metrics

The ECTS is a multi-objective problem involving two contradicting objectives accuracy and earliness. Accuracy Eq. (2) represents the percentage of true predictions out of total predictions. Earliness Eq. (3) presents the average predictive length of the time series. These metrics are formally defined as:

$$Accuracy = \frac{\text{Correct predictions}}{\text{Total predictions}} * 100 \tag{2}$$

$$Earliness = \frac{1}{N} \sum \frac{\text{Number of data points are used for making decision}}{\text{Total data points in the time series}} \tag{3}$$

where N is the number of time series in the dataset. A low value of Earliness represents that the model predicted the class at a very early stage whereas a higher value of earliness represents that the model predicted the class very late.

3.2 Dataset

The datasets used for evaluation are publicly available on UCR archive [19]. We used seven different datasets, four sensor type datasets *ItalyPowerDemand,*

[1] https://www2.informatik.hu-berlin.de/~schaefpa/teaser/.

SonyAIBORobotSurface1, *SonyAIBORobotSurface2* and *MoteStrain*, two ECG (Electrocardiogram) type datasets *ECG200* and *TwoLeadECG*, one motion type dataset *GunPoint* as shown in Table 1.

3.3 Dataset Preprocessing

Datasets are taken from UCR archive [19] which already provides a separate train and test set. We have not performed any pre-processing on time series. However, the proposed model accepts the fixed input length of time series. Therefore, to make the model adaptable to different length inputs, we used zero padding. For example, if the length of the complete time series is T and the model is trying to make a prediction at time-step t then $(T - t)$ data points are being padded with zero.

3.4 Results Analysis

The RCRL model has been evaluated on seven publically available datasets as shown in Table 1. The RCRL achieved good earliness comparable to other methods while maintaining acceptable accuracy. Table 2 presented the performance of RCRL by considering two kinds of RNN cells: LSTM and GRU. It is observed that both variants provided comparable results and none of them is superior. It depends on the sequence length, type of data the series has. The selection of λ helps to get desired earliness with acceptable accuracy.

3.5 Effect of Parameter λ

The RL agent maximizes its reward by selecting the optimal halting point for maximum accuracy. At some point during training, the agent learns that the halting criteria and receives maximum reward, probably at the completion of time series. To incorporate earliness into the RL agent, an additional term is being added to the loss function which promotes adaptive early halting based on the hyperparameter λ. The magnitude of λ determines the weightage of earliness for the RL agent, by increasing the λ. The effect of the λ parameter has been analyzed on accuracy and earliness for the "GunPoint" dataset, as demonstrated in Fig. 2. It can be seen that as the values of λ increases, the earliness improves and accuracy decreases. Moreover, the value of accuracy and earliness both increases with λ because of conflicts between these two objectives.

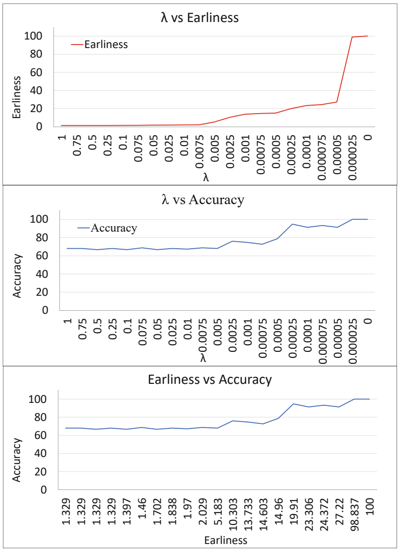

Fig. 2. Effect of hyperparameter λ.

Table 2. RCRL Accuracy

Dataset	*RCRL(LSTM)*			*RCRL(GRU)*		
	λ	*Acc %*	*Ear %*	λ	*Acc %*	*Ear %*
MoteStrain	0.0008	65	8.7	0.0008	80	10.99
SonyAIBORobotSurface1	0.0008	85	8.7	0.0008	85	9.5
TwoLeadECG	0.00001	80	21	0.0008	71	11.55
SonyAIBORobotSurface2	0.0008	78	15.9	0.0008	71	9.448
GunPoint	0.0008	87	16.1	0.0008	82.6	16.349
ItalyPowerDemand	0.02	67	19.63	0.0008	78.4	42.44
ECG200	0.0008	88	9.7	0.0008	86	10.84

3.6 Training Loss and Accuracy

The learning process of the RCRL has been analyzed on *GunPoint* dataset as shown in Fig. 3, First the classifier has been trained for 50 epochs as shown in Fig. 3(a) and 3(b). Figure 3(a) shows an epoch vs accuracy graph for train and test accuracy, while Fig. 3(b) shows an epoch vs test and train losses graph. Figure 3(c) and 3(d) represents the reinforcement learning agent's accuracy, earliness, and loss. As defined earlier the *Accuracy* and *Earliness* are contradictory terms, Fig. 3(c) demonstrates how the reinforcement learning agents balance the trade-off while training, and Fig. 3(d) shows an epoch vs loss graph with an additional term added to the loss function to penalise for late prediction.

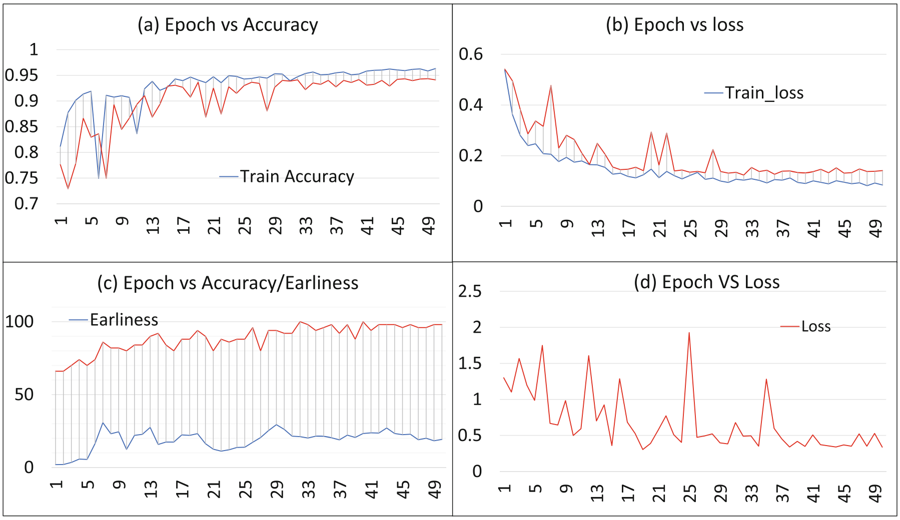

Fig. 3. (a) Epoch vs Accuracy of base classifier, (b) Epoch vs loss of base classifier, (c) Epoch vs Accuracy/Earliness of RL agent, (d) Epoch vs loss of RL agent.

4 Conclusion

We proposed the RCRL model for the early classification of time series. The RCRL utilizes the capabilities of Convolutional Neural Network, Recurrent Neural network, and Reinforcement Learning. The proposed RCRL extracts the features from incomplete time series using the proposed hybrid classifier and learns the decision policy with the help of a reinforcement learning agent. The RCRL model demonstrated the ability to balance the trade-off and give better performance than other alternative models. We compared RCRL with existing models and the results show that the RCRL model delivers state-of-the-art performance in terms of accuracy and earliness.

In future work, we will prepare a complex feature set to learn a more robust decision policy and optimize it for an application-specific problem.

References

1. Bagnall, A., Lines, J., Bostrom, A., Large, J., Keogh, E.: The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Min. Knowl. Disc. **31**(3), 606–660 (2016)
2. Rußwurm, M., Tavenard, R., Lefèvre, S., Körner, M.: Early classification for agricultural monitoring from satellite time series. arXiv preprint arXiv:1908.10283 (2019)
3. Hatami, N., Chira, C.: Classifiers with a reject option for early time-series classification. In: 2013 IEEE Symposium on Computational Intelligence and Ensemble Learning (CIEL), pp. 9–16. IEEE (2013)
4. Nath, A.G., Sharma, A., Udmale, S.S., Singh, S.K.: An early classification approach for improving structural rotor fault diagnosis. IEEE Trans. Instrum. Measur. **70**, 1–13 (2021)
5. Ghalwash, M.F., Ramljak, D., Obradović, Z.: Patient-specific early classification of multivariate observations. Int. J. Data Min. Bioinform. **11**(4), 392 (2015)
6. Sharma, A., Singh, S.K.: A novel approach for early malware detection. Trans. Emerg. Telecommun. Technol. (2020)
7. Sharma, A., Singh, S.K., Udmale, S.S., Singh, A.K., Singh, R.: Early transportation mode detection using smartphone sensing data. IEEE Sens. J. **21**, 15651–15659 (2020)
8. Mori, U., Mendiburu, A., Keogh, E., Lozano, J.A.: Reliable early classification of time series based on discriminating the classes over time. Data Min. Knowl. Discov. **31**(1), 233–263 (2017)
9. Hartvigsen, T., Sen, C., Kong, X., Rundensteiner, E.: Adaptive-halting policy network for early classification. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019. ACM Press (2019)
10. Hetland, M.L.: A survey of recent methods for efficient retrieval of similar time sequences. In: Data Mining in Time Series Databases, pp. 23–42. World Scientific (2004)
11. Xing, Z., Pei, J., Philip, S.Y.: Early classification on time series. Knowl. Inf. Syst. **31**(1), 105–127 (2011)

12. Mori, U., Mendiburu, A., Miranda, I.M., Lozano, J.A.: Early classification of time series using multi-objective optimization techniques. Inf. Sci. **492**, 204–218 (2019)
13. Lv, J., Xuegang, H., Li, L., Li, P.: An effective confidence-based early classification of time series. IEEE Access **7**, 96113–96124 (2019)
14. He, G., Zhao, W., Xia, X., Peng, R., Wu, X.: An ensemble of shapelet-based classifiers on inter-class and intra-class imbalanced multivariate time series at the early stage. Soft. Comput. **23**(15), 6097–6114 (2018). https://doi.org/10.1007/s00500-018-3261-3
15. Xing, Z., Pei, J., Yu, P.S., Wang, K.: Extracting interpretable features for early classification on time series. In: Proceedings of the 2011 SIAM International Conference on Data Mining, pp. 247–258. SIAM (2011)
16. Anderson, H.S., Parrish, N., Tsukida, K., Gupta, M.R.: Reliable early classification of time series. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE (2012)
17. Karim, F., Majumdar, S., Darabi, H., Harford, S.: Multivariate LSTM-FCNs for time series classification. Neural Netw. **116**, 237–245 (2019)
18. Flores, C., Taramasco, C., Lagos, M.E., Rimassa, C., Figueroa, R.: A feature-based analysis for time-series classification of Covid-19 incidence in Chile: a case study. Appl. Sci. **11**(15) (2021)
19. Dau, H.A., et al.: The UCR time series archive. CoRR, abs/1810.07758 (2018)
20. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. CoRR, abs/1709.01507 (2017)
21. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
22. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling (2014)
23. Schäfer, P., Leser, U.: Teaser: early and accurate time series classification. Data Min. Knowl. Disc. **34**(5), 1336–1362 (2020)
24. Parrish, N., Anderson, H.S., Gupta, M.R., Hsiao, D.Y.: Classifying with confidence from incomplete information. J. Mach. Learn. Res. **14**(1), 3561–3589 (2013)