Game Theory based Early Classification of Rivers using Time Series Data

Ashish Gupta[1], Rajdeep Pal[2], Rahul Mishra[1],
Hari Prabhat Gupta[1], Tanima Dutta[1], and Priyank Hirani[3]
{ashishg.rs.cse16, rajdeep.pal.ece16, rahulmishra.rs.cse17, hariprabhat.cse, tanima.cse}@iitbhu.ac.in, phirani.iic@gmail.com
[1]Department of CSE, IIT (BHU) Varanasi, [2]Department of Electronics Engineering, IIT (BHU) Varanasi,
[3]University of Chicago Trust, India.

***Abstract*—An important issue in water quality monitoring of rivers is to identify the river by using the time series data. Water quality involves parameters such as pH, electrical conductivity, dissolved oxygen, turbidity, *etc*. For real-time monitoring, such data are collected using sensors from various geographical locations, in different seasons. In this work, our objective to classify the rivers using time series of the parameters, as early as possible. It is difficult to assign the correct label to the river if a time series does not have any information on geographical location. Therefore, we propose an early classification approach to classify the rivers using time series data. We use a probabilistic classifier to obtain the posterior class probabilities to state a tradeoff between accuracy and earliness. A game model is developed to find the optimal number of data points that can provide the desired level of accuracy. We evaluate the proposed approach by classifying the three major rivers of India using time series data. The experimental results illustrate that the proposed approach performs better than the existing approaches in terms of accuracy and earliness.**

***Index Terms*—Early classification, game theory, rivers, time series.**

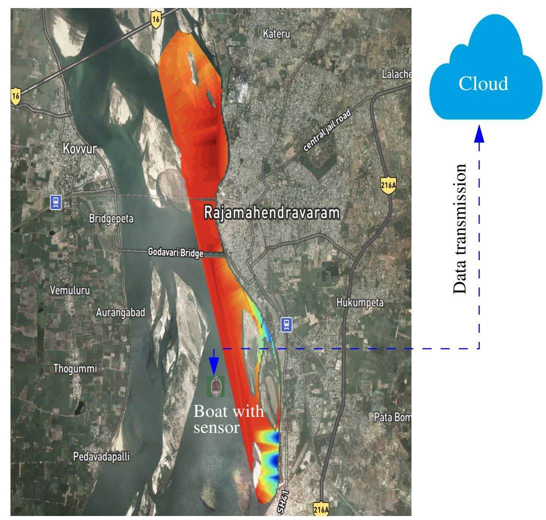


Fig. 1: Illustration of acquisition of data using mobile sensors.

I. Introduction

Real-time monitoring of a water body using sensors is rapidly growing day-by-day due to easier and cheaper availability of the sensors. Water body, such as rivers are usually monitored by measuring various parameters including pH value, dissolved oxygen, electrical conductivity, temperature, *etc*. These parameters can help to identify the pollution level of the river water. In traditional methods, the rivers are usually monitored from the samples of water taken from discrete sites of the river [1]. Such methods can not describe the temporal and spatial changes in parameters of the streaming water. Therefore, for real-time monitoring, the mobile sensors take a sequence of measurements over a stretch of the river using a boat at an equal sampling rate. Fig. 1 illustrates a scenario where mobile sensors are used to acquire data and transfer to the Cloud. This sequence of temporal measurements is known as time series [2]. The time series consists of the measurements of any particular parameter (*e.g.,* pH, dissolved oxygen, electrical conductivity, *etc*.).

Time Series Classification (TSC) is a supervised learning method that predicts the class label of an unseen time series using a given labeled dataset. It learns a mapping from time series to class label using labeled dataset during training. There exists a wide range of Internet of Things (IoT) applications of TSC that include environmental monitoring [3], health monitoring [4], and smart agricultural [5]. Li *et al.* [3] identify the chlorine concentration in the drinking water using TSC. In [6], the authors developed a TSC approach to classify the sleep stages based on EEG signals. Fazeen *et al.* [7] classify the driver's behavior using smartphone sensors for safe driving.

Most of the IoT applications require the classification of the given time series within the maximum allowable response time. Such response time is known as the deadline of the application. It is desirable to classify an incoming time series as soon as possible using minimum available data points. Traditional TSC approaches [2], [3], [6] use all the data points to classify the time series. The time complexity of the existing approaches is high and therefore not suitable for IoT applications, where applications have deadline constraints. The number of data points, that are used in the classification, specifies how early an incoming time series is classified. Such classification is known as *early classification* of time series in the literature [8]. Earliness can only be achieved if the

978-1-5386-4980-0/19/$31.00 ©2019 IEEE

classification approach does not wait for all the data points of an incoming time series. A time series can be classified as accurately as possible if a classifier uses all the data points, but such classifier does not provide any earliness and therefore not suitable for IoT applications. It clearly indicates that there exists a tradeoff between accuracy and earliness.

The main objective of an early classification approach is to find Minimum Required Length (MRL) of time series for classification while maintaining a desired level of accuracy. Such desired level of accuracy, denoted by $\boldsymbol{\alpha}$, less than or equals to unity. Ghalwash *et al.* [9] proposed a shapelet based early classification for classifying the crisis in the stock market. In [10], the authors use early classification for identifying the faults that can occur in an industrial plant. In any running industrial process, even a small fault can lead to bigger losses. So, the source of fault must be identified as early as possible. In [11], an early classification approach is presented for differentiating the wind noise from the bird's song. This early classification of the bird songs leads to the battery saving of the recording device. The authors in [12] developed an early identification system to monitor the chemical leaks using the time series of sensory odor signals.

In this paper, we address the problem: *how to predict the label (or name) of a river using an incoming time series of sensory data of water parameters as early as possible?* We assume that the incoming time series does not have the information about the label or name of the river. To address this problem, we first select one parameter from the set parameters that can provide sufficient identifiable information about the river. Later, we use the time series of the selected parameter to classify the river using Gaussian Process (GP) classifier [13]. Further, we define a game model to obtain the optimum MRL of time series for early classification of the rivers with desired level of accuracy $\boldsymbol{\alpha}$.

• **Motivation and Major Contributions:** To the best of our knowledge, we are the first to address the early classification problem for water body using game theory. We develop a game model to estimate the MRL while maintaining the desired level of accuracy $\boldsymbol{\alpha}$, using given labeled dataset. The motivation and major contributions of this work are as follows:

- The existing approaches [14], [15] of early classification use heuristic based algorithms for optimizing the trade-off between accuracy and earliness while others [9], [10] do not consider any optimization in the classification. Such trade-off can better be modeled using game theory [16]. We propose a game theory based solution for obtaining the MRL while maintaining the desired level of accuracy of the classifier. The advantages of game theory as compared to heuristic-based approaches are that the game theory requires less time and computations.
- In a water-to-cloud data transmission[1], sensory data of the water parameter including GPS coordinates, is transmitted to the cloud using IoT solutions. The GPS coordinates are used to assign the label to the different rivers. If the GPS coordinates get lost or not transmitted properly to the cloud then it becomes difficult to assign the correct label to the river. Therefore, we present an early classification approach that can classify a river using an incoming time series of its parameters without the GPS coordinates. The less data transmission reduces the energy consumption and delay of the IoT solutions.
- As the use of GPS device consumes large amount energy in data collection [17], we only collect the sensory data of various parameters of the rivers and propose an approach to identify the rivers using the characteristics of streaming water.

The rest of the paper is organized as follows: In next section, we discuss the proposed early classification approach using game model. Section III illustrates the experimental results on river dataset[1] and compare the proposed approach with existing approaches. Section IV concludes the paper by providing the motivation for future research in this direction.

II. Early Classification using Game Theory

In this section, we propose an early classification approach for time series dataset using game theory. We assume that the classification is performed on the Cloud at high end machines. The dataset for early classification of the rivers are collected by using mobile sensors. Such sensors are attached with the boat to collect the particular parameter (*e.g.,* pH, dissolved oxygen, electrical conductivity, *etc.*). We prepared a road-map across the rivers to ensure maximum possible data collection. We repeated the data collection step regularly to identify the temporal changes of the parameter. Finally, we transferred the sensory data to the Cloud using IoT solutions for classification. The proposed early classification approach consists training phase and testing phase. In training phase, an early classifier is constructed by using a GP classifier and a game model. In testing phase, we use this built early classifier to classify an incoming time series.

A. Training phase

The main objective of this phase is to construct an early classifier $\mathbf{H}$ using a collected labeled river dataset. In this phase, we first use GP classifier to compute the posterior class probabilities, which are used to state the tradeoff between earliness and accuracy. Later, we propose a two players game model to estimate the MRL that optimizes the stated tradeoff. Let $\mathbf{D}$ is a given labeled dataset which consists of N time series and each time series has M data points. Let l denotes the number of class labels in $\mathbf{D}$ and L_q represents a class label of any time series, where $1 \leq q \leq l$. Fig. 2 illustrates the complete procedure of training phase.

1) GP classifier for formulating optimization problem: In this step, we use GP classifier for estimating the MRL of each time series as it works well in similar framework [14]. The GP is a probabilistic classifier that uses Bayes rule to compute the posterior class probabilities for each time series of given dataset $\mathbf{D}$. The posterior probability of a time series

[1] The complete information about the collection of river dataset is available at www.thoreau.uchicago.edu./water_body_overview

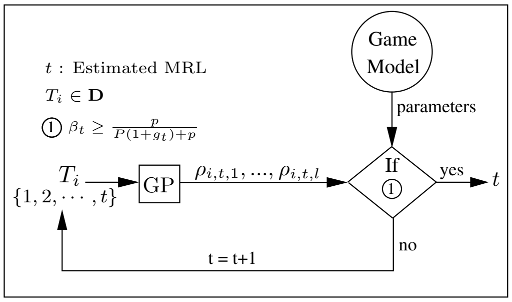


Fig. 2: Illustration of training phase of the proposed classifier.

$T = \{t_1, t_2, \cdots, t_M\}, \forall t_j \in \mathbb{R}$, for any L_q class label, can be computed as

$$P\left(\frac{L_q}{T}\right) = \prod_{j=1}^{M} \frac{P(\frac{t_j}{L_q})P(L_q)}{P(t_j)}, \tag{1}$$

where $P(L_q)$ and $P(t_j)$ are prior and marginal probabilities that can be computed using $\mathbf{D}$. The likelihood term $P(\frac{t_j}{L_q})$ can be obtained as

$$P\left(\frac{t_j}{L_q}\right) = \frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-\frac{(t_j-\mu)}{2\sigma^2}},\ 1 \leq j \leq M, \tag{2}$$

where μ and σ denote the mean and standard deviation of time series T. Let $\rho_{i,M,q}$ and $\rho_{i,t,q}$ denote the posterior class probability of time series $T_i \in \mathbf{D}$ for class L_q using M and t data points, respectively. The probabilities $\rho_{i,M,q}$ and $\rho_{i,t,q}$ can be computed using Eq. 1. Now, the MRL of a time series T_i is t, if the following inequality holds.

$$\boldsymbol{\alpha} \times \rho_{i,M,q} \leq \rho_{i,t,q}, \tag{3}$$

where $\boldsymbol{\alpha}$ denotes the desired level of accuracy. From Eq. 3, it is obvious that the early classification involves two conflicting objectives which should be optimized simultaneously, *i.e.,* minimizing t and maximizing $\boldsymbol{\alpha}$. This optimization problem can be solved using game theory.

2) Game model for solving optimization problem: In this step, we formulate a two players game where accuracy and earliness serve as players. Let A denotes the accuracy as player 1 and E denotes the earliness as player 2. The goal of player 1 is to maximize the accuracy of the prediction whereas player 2 tries to minimize the number of data points used in the prediction of the class label. We assume that the aim of both players is to maximize their payoffs by choosing the best response against the strategy chosen by opponent. Each player can choose one of the two possible actions: now (n) and wait (w). The action n indicates that the classifier should predict the class label immediately whereas w indicates that the classifier should delay the prediction, by waiting for more data points of the time series. The game model can be define as

$$\text{Player(s)} \Rightarrow \begin{cases} Accuracy(A) & \text{Player 1} \\ Earliness(E) & \text{Player 2} \end{cases}$$

$$\text{Action(s)} \Rightarrow \begin{cases} now(n) & \text{Action 1} \\ wait(w) & \text{Action 2} \end{cases}$$

There are four possible combinations of strategies given as $(n, n), (n, w), (w, n)$, and (w, w), where first and second strategies are chosen by player A and E, respectively. We first define the notations that are used in the payoff functions as follows:

c_t	Delay cost to be paid by player E for not making the prediction at time step t and delaying it to next step $t+1$.
P	Profit for making correct prediction.
p	Loss incurred due to incorrect prediction.
R	Running cost of the sensor per unit time.
h_t	A classifier that is trained over time series dataset using first t data points, *i.e,* $h_t : \mathbb{R}^t \rightarrow L_q$, where $L_q \in \{L_1, L_2, \cdots, L_l\}$.
β_t	Success rate of h_t using cross-validation.
g_t	Probability of correct prediction at some later time step $t' > t$ and $t' \leq M$.

TABLE I: Payoff matrix with players and actions.

Accuracy (A) \ Earliness (E)	**now (n)**	**wait (w)**
now (n)	$\beta_t P$, $c_t + (M-t)R$	0, $-\infty$
wait (w)	$-\beta_t g_t P + (1-\beta_t)p$, $c_t + (M-t)R + \beta_t P - (1-\beta_t)p$	$g_t P$, $-c_t$

Next, we define payoff functions for both the players for each combination of strategies, which are illustrated in Table I. The cell on top left represents that if A and E both decide strategy n using t data points then A achieves a payoff of $\beta_t P$ and E gets a payoff of $(M-t)R + c_t$. Here, E saves $M-t$ number of data points, which will reduce the running cost $(M-t)R$ of the sensor. In addition, a delay cost c_t will also be added as the payoff of player E. Thus, the total payoff of E becomes $(M-t)R + c_t$. If A chooses n and E chooses w using t data points then the payoff achieved by the player A is 0, which indicates that there will be no substantial betterment in prediction accuracy at later step. The decision of E to wait for more data points does not make any sense if the desired level of accuracy is already achieved. So, this decision will incur very high penalty on E, which is indicate by $-\infty$ as the payoff of E.

In Table I, the bottom left cell represents the case where A selects strategy w and E selects n. In this case, A achieves the payoff $-\beta_t g_t P + (1-\beta_t)p$ and E gets the payoff $c(t) + (M-t)R + \beta_t P - (1-\beta_t)p$. If β_t is high then E has more chance for making correct prediction but, strategy A to wait seems useless for early classification which will incur more penalty on A as $-\beta_t g_t P$. The term $(1-\beta_t)$ indicates that if classifier h_t has poor success rate then decision of player A is dominating. Thus, a profit of $(1-\beta_t)p$ will be added in payoff of A. In the payoff of E, first two terms are already explained. Now, last term $(1-\beta_t)p$ is a penalty incurred for making late decision. At last, we explain the last combination of strategies where player A and E both select the same strategy w. Here, player A will get benefited as the accuracy of prediction gets better as the number of data points increases. So, the payoff of A is g_tP. Now, payoff of E is $-c_t$, which indicates the penalty of delay cost c_t for not making decision using t data points.

Definition 1. *The best response of player i upon selecting any strategy s_j by the player j, is a strategy $s_i^* \in S_i$ (where S_i is the set of strategies of player i), that satisfies the following condition:*

$$\begin{aligned} \textit{payoff}_i(s_i^*, s_j) &\geq \textit{payoff}_i(s_i, s_j), \\ \textit{where } s_i \in S_i &\textit{ and } s_j \in S_j. \end{aligned}$$

• **Selection of best response:** Here, we determine the best response for each player against the strategies chosen by other player. For determining the best response of the player A, we need to identify the strategies (*i.e.,* n or w) selected by player E and vice-versa. Let u_A and u_E denote the utility of player A and E, respectively. The process of determining best responses of both players will form following cases:

• (n, n): If A fixes its strategy as n then E can take either n or w, which is given as

$$\begin{aligned} u_E(n, n) &\geq u_E(n, w), \\ c_t + (M-t)R &\geq -\infty. \end{aligned} \tag{4}$$

Similarly, when E fixes its strategy n then A can have two possible strategies n or w, which will hold following condition:

$$\begin{aligned} u_A(n, n) &\geq u_A(w, n), \\ \beta_t &\geq \frac{p}{P(1+g_t)+p}. \end{aligned} \tag{5}$$

• (n, w): When player A fixes its strategy as n,

$$\begin{aligned} u_E(n, w) &\geq u_E(n, n), \\ -\infty &\geq c_t + (M-t)R. \end{aligned} \tag{6}$$

The above condition will never be true as $c_t+(M-t)R > 0$. In early classification, if desired level of accuracy is achieved then waiting for more data points is never advisable. Hence, this case can be discarded from best responses.

• (w, n): If A fixes its strategy as w, we get

$$\beta_t \geq \frac{p - (2c_t + (M-t))R}{P+p}. \tag{7}$$

If player E decides to take strategy n, we get

$$\beta_t \leq \frac{p}{P(1+g_t)+p}. \tag{8}$$

From Eq. 7 and Eq. 8, we obtain an expression for β_t as

$$\frac{p - (2c_t + (M-t))R}{P+p} \leq \beta_t \leq \frac{p}{P(1+g_t)+p}. \tag{9}$$

• (w, w): Similar to above cases, when A fixes its strategy w, we get

$$\beta_t \leq \frac{p - (2c_t + (M-t)R)}{P+p}. \tag{10}$$

Further, when E fixes its strategy w, we get $g_tP > 0$, which is always true.

From the above four cases, we observe that the success rate β_t is highest at (n, n) among all the cases, which is always desirable. We therefore conclude that the best response of both the players can be achieved at (n, n). At this end, the value of t that satisfies $\beta_t = \frac{p}{P(1+g_t)+p}$ will be the MRL of the considered time series.

Once we obtain the MRL for each time series of dataset $\mathbf{D}$, we compute a class representative MRL for all the class labels. Let A_q is an array of the obtained MRL of the time series that belong to class label L_q, where $1 \leq q \leq l$. We calculate the Inter Quartile Range (IQR) as $Q_3 - Q_1$, where Q_1 and Q_3 denote the 25 and 75 percentile of A_q, respectively. Now, we remove the MRLs from A_q that lie outside the range from $Q_1 - 1.5 \times$ IQR to $Q_3 + 1.5 \times$ IQR. Later, we take maximum value of MRL in A_q as the representative MRL for class L_q. Finally, an early classifier $\mathbf{H}$ is build that consists of l classifiers using the class representative MRL.

B. Testing phase

In this phase, we predict the class label of an incoming time series using the built early classifier $\mathbf{H}$ while maintaining the desired level of accuracy $\boldsymbol{\alpha}$. Let T' is an incoming time series which is to be classified and t' is the number of arrived data points. Now, the objective of the classifier $\mathbf{H}$ is to check whether t' data points of T' are sufficient for prediction or not. If the arrived data points of T' are equal to the estimated MRL then class label (*i.e.,* L_q) of the incoming time series T' can be predicted using GP classifier, else the early classifier will wait for more data points. The block diagram of the complete procedure of testing phase is shown in Fig. 3.

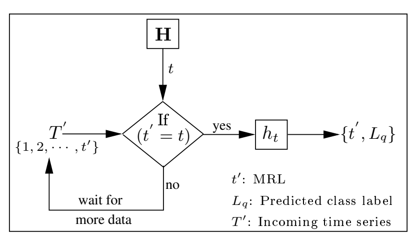


Fig. 3: Illustration of testing phase of the proposed classifier.

III. EARLY CLASSIFICATION OF RIVERS AND RESULTS

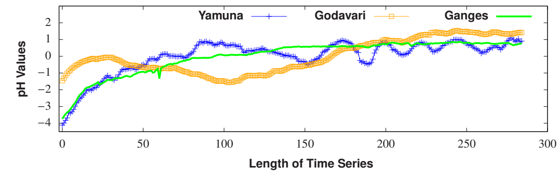
In this section, we evaluate the proposed approach by classifying the major rivers of India using time series data. We first explain the river dataset and the use this dataset to predict the class label of an incoming time series of water parameter. Finally, we compare the proposed approach with existing approaches [14], [15].

A. River dataset

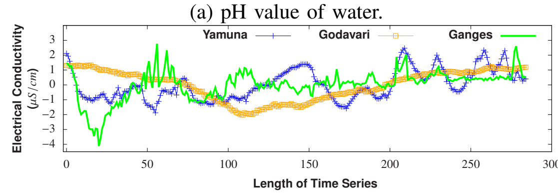
This dataset is created to monitor the water pollution in major rivers of India that include Yamuna, Godavari, and Ganges. The objective is to study the impact of water pollution on health of people who live near the coasts of the rivers. The major causes for the pollution are agricultural wastes, industrial water drainage, and anthropogenic activities. The parameters, that are considered to measure the water quality, are temperature, pH, Electrical Conductivity (EC), Dissolved Oxygen (DO), total dissolved solid, turbidity, ammonia, and *etc.* Water quality data is collected using a boat which is attached with sensor. Hanna multiparameter HI-9829 sensor is used to take the time series measurements on a predefined route of the river. The data that is obtained from the water immersed sensor probe, is recorded on a hand held device. For each river, many boat rides are executed to collect temporal and spatial data for above mentioned parameters at different locations across the river, in different seasons. This data is later uploaded to a cloud server. Table II summarizes the river dataset. In this work, we use this dataset to build a classifier that can predict a label of an incoming time series of the river.

TABLE II: Summary of river dataset.

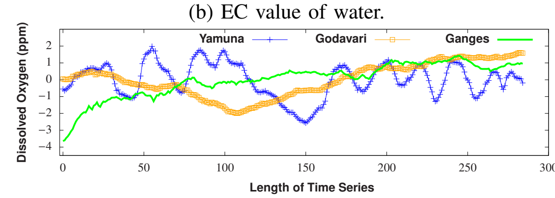
	Locations	Length of time series	Number of time series
Yamuna	Delhi	$368-12140$	21
Godavari	Rajahmundry, Narsapur	$459-13625$	136
Ganges	Kolkata, Varanasi, Prayagraj	$285-1262$	30



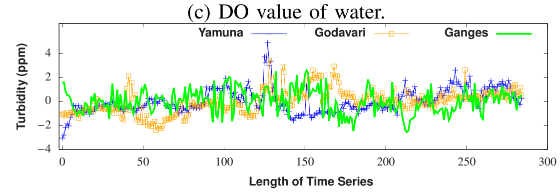
(a) pH value of water.



(b) EC value of water.



(c) DO value of water.



(d) Turbidity value of water.

Fig. 4: Illustration of mean time series of the four parameters (pH, EC, DO, and Turbidity) for Yamuna, Godavari, and Ganges rivers.

B. Selection of most identifiable parameter

In this step, our objective to find the most identifiable parameter among all the parameters. As the data is not available for some of the parameters for all the rivers, we consider the only following parameters pH, EC, DO, and Turbidity, in our experiment. Fig. 4 illustrates the mean time series of the considered four parameters for three rivers (*i.e,* Yamuna, Godavari, and Ganges). Here, we consider the shortest time series (*i.e.,* 285 data points) from each river. It can be clearly seen from part (a) and (c) of Fig. 4, that the time series of pH and DO parameters contain most distinguishing patterns for river identification. Moreover, pH time series has the river identifiable pattterns earlier than DO. Therefore, we select pH time series to evaluate the proposed approach.

C. Experimental Results

We evaluate the proposed early classification approach using only pH time series. Let $\mathbf{D}$ denotes the dataset that consists of only pH time series of the rivers. It consists of 187 time series and each time series has 285 data points, *i.e.,* $N = 187$ and $M = 285$. We split the dataset $\mathbf{D}$ into training and testing dataset with 70% and 30% time series, respectively. Now, we set the parameters for game model considering that the classification task is moderately important, as profit $P = 3$, penalty $p = 3$, and running cost of sensor per unit time $R = 3$. The delay cost c_t is taken as quadratic function of t, where t is the time step at which the decision is being made. Further, parameter g_t and β_t are learned from the training dataset. We consider two evaluation metrics: accuracy and earliness. Accuracy is computed as the ratio of correctly predicted to total number of time series in the testing dataset. Finally, we compute the earliness as

$$\text{Earliness}(\%) = \frac{M - t}{M} \times 100. \tag{11}$$

where t denotes the MRL of time series.

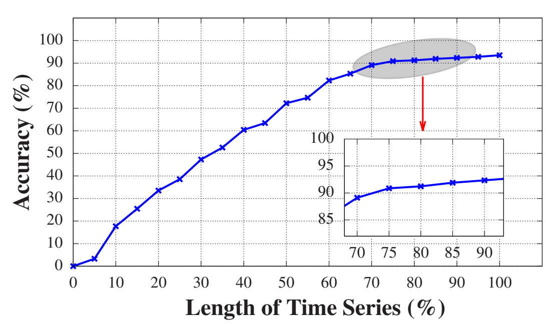


Fig. 5: Illustration of tradeoff between accuracy and earliness along the progress of the time series of pH parameter.

We first evaluate the proposed approach along the progress of time series with an interval of 5%. Fig. 5 demonstrates the obtained graph of tradeoff between accuracy and earliness using pH time series testing data. It is clear from the graph that initially the accuracy increases very fast and reaches to around 88% using only 70% of the time series. Later, the accuracy changes marginally by using the remaining data points (*i.e.,* 70% to 100%), as shown by grey ellipse in Fig. 5. It indicates that early decision can be made without waiting for full length time series with a small compromise of accuracy.

In addition, we also present the confusion matrix at desired level of accuracy $\boldsymbol{\alpha} = 0.8$ and $\boldsymbol{\alpha} = 0.9$, as shown in Table III. It can be clearly observe from the table that the proposed approach is able to achieve desired level of accuracy for Yamuna and Godavari rivers at both $\boldsymbol{\alpha} = 0.8$ and $\boldsymbol{\alpha} = 0.9$.

TABLE III: Confusion matrix of the proposed approach.

	$\boldsymbol{\alpha} = 0.8$			$\boldsymbol{\alpha} = 0.9$		
Rivers	Yamuna	Godavari	Ganges	Yamuna	Godavari	Ganges
Accuracy	82.20%	83.40%	76.50%	88.15%	90.15%	85.10%

Finally, we compare the proposed approach with two existing approaches including ECDIRE [14] and Optimizing Accuracy and Earliness (OAE) [15]. Table IV illustrates the comparison results on accuracy and earliness. It is clear from the results that the proposed approach outperforms the existing approaches on both the metrics. ECDIRE performs better than OAE on accuracy but worse on earliness. As OAE uses a heuristic based optimization algorithm, it provides better earliness than ECDIRE. The proposed approach is able to classify the rivers at the desired level of accuracy with a marginal difference of $(90\% - 87.80\%) = 2.20\%$ at $\boldsymbol{\alpha} = 0.9$, and with the maximum earliness (*i.e.,* 39.35%) at $\boldsymbol{\alpha} = 0.8$.

TABLE IV: Comparison of the proposed approach with existing approaches on accuracy and earliness.

	$\boldsymbol{\alpha} = 0.8$		$\boldsymbol{\alpha} = 0.9$	
	Accuracy (%)	Earliness (%)	Accuracy (%)	Earliness (%)
ECDIRE	67.65	20.60	70.30	15.45
OAE	65.55	25.50	68.65	22.30
Proposed	**80.70**	**39.35**	**87.80**	**32.60**

IV. Conclusion

In this paper, we propose a Game theory based early classification approach to classify the rivers, by using only sensory data of water parameters. Such classification of rivers is required when time series are unlabeled or not tagged with GPS coordinates. We employ a Gaussian Process classifier to obtain the posterior class probabilities using the river dataset. The main objective of the early classification approach is to find the minimum number of data points that can correctly classify an incoming time series with a small compromise of accuracy. It clearly indicates that there exists a tradeoff between accuracy and earliness. We, therefore, build a game model to optimize this tradeoff.

The experimental results showed that the proposed approach is able the classify the rivers as early as possible. The confusion matrix illustrated the highest accuracy for Godavari river. Finally, the comparison table showed that the proposed approach gains on accuracy and earliness over the existing approaches. This work motivates the further research on identification of the pollution sources across the river using only the time series data of various parameters. We believe that the proposed classification approach can also be used in other IoT applications.

Acknowledgements

We would like to thank Professor Supratik Guha for giving his valuable suggestions. This study is sponsored by the Tata Centre for Development at UChicago.

References

[1] H. Goncalves, A. C. Teodoro, and H. Almeida, "Identification, characterization and analysis of the douro river plume from meris data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 5, pp. 1553–1563, 2012.

[2] A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh, "The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances," *Data Mining and Knowledge Discovery*, vol. 31, no. 3, pp. 606–660, 2017.

[3] L. Li, J. McCann, N. S. Pollard, and C. Faloutsos, "Dynammo: Mining and summarization of coevolving sequences with missing values," in *Proceedings of ACM SIGKDD*, 2009, pp. 507–516.

[4] U. Satija, B. Ramkumar, and M. S. Manikandan, "Real-time signal quality-aware ecg telemetry system for iot-based health care monitoring," *IEEE Internet of Things Journal*, vol. 4, no. 3, pp. 815–823, 2017.

[5] S. Heble, A. Kumar, K. V. V. D. Prasad, S. Samirana, P. Rajalakshmi, and U. B. Desai, "A low power iot network for smart agriculture," in *Proceedings of IEEE WF-IoT*, 2018, pp. 609–614.

[6] M. Diykh, Y. Li, and P. Wen, "Eeg sleep stages classification based on time domain features and structural graph similarity," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 11, pp. 1159–1168, 2016.

[7] M. Fazeen, B. Gozick, R. Dantu, M. Bhukhiya, and M. C. González, "Safe driving using mobile phones," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1462–1468, 2012.

[8] U. Mori, A. Mendiburu, S. Dasgupta, and J. A. Lozano, "Early classification of time series by simultaneously optimizing the accuracy and earliness," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4569–4578, 2018.

[9] M. F. Ghalwash, V. Radosavljevic, and Z. Obradovic, "Utilizing temporal patterns for estimating uncertainty in interpretable early decision making," in *Proceedings of ACM SIGKDD*, 2014, pp. 402–411.

[10] A. Bregón, M. A. Simón, J. J. Rodríguez, C. Alonso, B. Pulido, and I. Moro, "Early fault classification in dynamic systems using case-based reasoning," in *Conference of the Spanish Association for Artificial Intelligence*, 2005, pp. 211–220.

[11] U. Mori, A. Mendiburu, S. Dasgupta, and J. A. Lozano, "Early classification of time series from a minimization point of view," in *Proceedings of NIPS*, 2015, pp. 1–5.

[12] N. Hatami and C. Chira, "Classifiers with a reject option for early time-series classification," in *Proceedings of CIEL*, 2013, pp. 9–16.

[13] C. E. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[14] U. Mori, A. Mendiburu, E. Keogh, and J. A. Lozano, "Reliable early classification of time series based on discriminating the classes over time," *Data Mining and Knowledge Discovery*, vol. 31, no. 1, pp. 233–263, 2016.

[15] U. Mori, A. Mendiburu, S. Dasgupta, and J. A. Lozano, "Early Classification of Time Series by Simultaneously Optimizing the Accuracy and Earliness," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 1–10, 2017.

[16] M. J. Osborne *et al.*, *An introduction to game theory*. Oxford university press New York, 2004.

[17] J. Liu, B. Priyantha, T. Hart, Y. Jin, W. Lee, V. Raghunathan, H. S. Ramos, and Q. Wang, "Co-gps: Energy efficient gps sensing with cloud offloading," *IEEE Transactions on Mobile Computing*, vol. 15, no. 6, pp. 1348–1361, 2016.