Early Classification of Multivariate Time Series with Different Sampling Rates

Report submitted in fulfillment of the requirements
for the Exploratory Project of

Second Year B.Tech.

by

Nishant Mittal

Under the guidance of

Dr. Hariprabhat Gupta

Department of Computer Science and Engineering
INDIAN INSTITUTE OF TECHNOLOGY (BHU) VARANASI
Varanasi 221005, India
May 2020

Dedicated to

My parents, professors, exploratory project convener, mentor and everyone who helped and motivated me in successful completion of this project.

Declaration

I certify that

1. The work contained in this report is original and has been done by myself and the general supervision of my supervisor.
2. The work has not been submitted for any project.
3. Whenever I have used materials (data, theoretical analysis, results) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.
4. Whenever I have quoted written materials from other sources, I have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

Place: IIT (BHU) Varanasi
Date:

Nishant Mittal
IDD Student
Department of Computer Science and Engineering,
Indian Institute of Technology (BHU) Varanasi,
Varanasi, INDIA 221005.

Certificate

*This is to certify that the work contained in this report entitled "**Early Classification of Multivariate Time Series with Different Sampling Rates**" being submitted by **Nishant Mittal** (**Roll No. 18074013**), carried out in the Department of Computer Science and Engineering, Indian Institute of Technology (BHU) Varanasi, is a bona fide work of our supervision.*

Place: IIT (BHU) Varanasi
Date:

Dr. Hariprabhat Gupta
Department of Computer Science and Engineering,
Indian Institute of Technology (BHU) Varanasi,
Varanasi, INDIA 221005.

Acknowledgments

I want to express my sincere gratitude to the people who have helped me the most throughout my project. I am grateful to my project supervisor **Dr. Hari Prabhat Gupta** for providing me an opportunity to implement the paper entitled "**Early Classification of Multivariate Time Series with Different Sampling Rates**" and his constant support for the project. I wish to thank my parents for their support and attention. I would like to thank my friends who encouraged us and helped us out in finalizing the project. At last, we also thank the Almighty for his blessings.

Place: IIT (BHU) Varanasi
Date: **Nishant Mittal**

Contents

List of Figures

List of Symbols

Symbol	Description
α	The accuracy to be maintained while the classification.
D	A labeled dataset.
N	Total number of MTD in the dataset .
C^i	Data of the i^{th} component(sensor).
n	Total number of sensors.
h_i	Constructed classifier for C^i component.
l	No of class labels in dataset D.
C^i_j	j^{th} MTD in C^i component.

Chapter 1

Introduction

1.1 Overview

Classification of Multivariate Time Series (MTS) data has been an important area of research for many years. In time-critical applications, such as health informatics, fire detection, and disaster forecasting, it is desirable to classify the MTS data as early as possible. This work proposes an early classification approach to classify an incoming MTS. The early classification approach helps to predict the class label of an incoming MTS without waiting for the full length. Different from the existing work, this work considers that sampling rate of the sensors which generated the MTS is different. The performance of the approach is evaluated on a publicly available dataset using accuracy, earliness and energy consumption.

Chapter 2

Project Work

2.1 Preliminaries

2.1.1 Assumptions

We will be assuming that our application of IoT consist of n sensors.All of these sensors which are going to constitute our MTS, are having different sampling rate.All these sensors collect data after a certain interval of time.Let us assume the full length of our time series be T. The MTS which consist of T data points in all its time series is known as complete MTS.

2.1.2 Component

The time series which is generated by a specific sensor over a fixed period of time is referred as a component of the MTS. It can also be referred as an ordered sequence of sensory data points.If we have n sensors then our MTS generated would consist of n components. Such MTS is denoted as $C = \{C_1, C_2, \ldots, C_n\}$ where C_i denotes ith component.

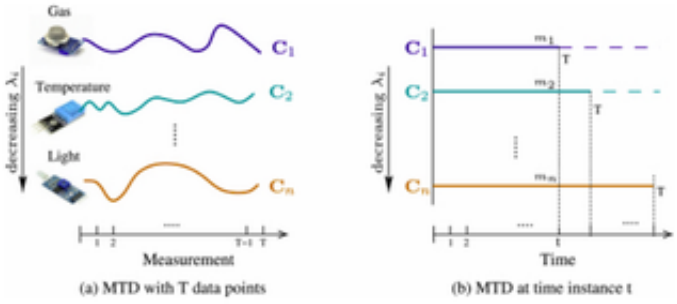

Figure 2.1 Illustration of an MTD with n components that are generated by various sensors.

2.1.3 Earliness

Let the full length of a component in the given MTD is T .The MRD is the number of minimum required data points to classify a component with α accuracy. The earliness of the component is defined as

$$\text{Earliness} = \frac{T-MRD}{T} \times 100$$

2.1.4 Accuracy

It refers to the percentage of correctly classified MTDs using class-wise MRD (with α desired level of accuracy). Let N denotes the number of correctly classified MTD in a given dataset with N MTD. Now, the accuracy can be mathematically expressed as

$$\text{Accuracy} = \frac{N'}{N} \times 100$$

2.1.5 Gaussian Process Classifier

A Gaussian Process (GP) [20] is an infinite set of space or time ordered random variables, where every finite set follows a multivariate normal distribution. The GP follows the joint distribution of those infinite random variables over continuous domain functions. It is a non-parametric classifier that models a time series as a finite set of random variables, which are the outputs of a stochastic process. Let X = $\{X[i] : 1 \leq t \leq T\}$ is an MTS of dataset D with where n is number of components in X.

2.1.6 Hidden Markov Model

Hidden Markov Model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process – call it X – with unobservable ("hidden") states. HMM assumes that there is another process Y whose behavior "depends" on X . The goal is to learn about X by observing Y . HMM stipulates that, for each time instance n_0, the conditional probability distribution of Y_{n_0} given the history $\{X_n = x_n\}_{n \leq n_0}$ must NOT depend on $\{x_n\}_{n<n_0}$.

2.1.7 Viterbi Algorithm

The Viterbi algorithm is a dynamic programming algorithm for finding the most likely sequence of hidden states—called the Viterbi path—that results in a sequence of observed events, especially in the context of Markov information sources and hidden Markov models (HMM).

2.1.8 Minimum required data points (MRL)

MRL is the minimum required data points of a component of a particular class label for maintaining at least α fraction of accuracy.

2.2 Datasets used to implement the approach

2.2.1 Dataset

I've used PEMS-SF (https://archive.ics.uci.edu/ml/datasets/PEMS-SF) dataset for this project. This dataset gives the readings by various sensors deployed on the roads and the day they were recorded on. We will use the readings by different sensors to predict the day they were recorded on.

2.3 Approach

2.3.1 Estimate Minimum Required Data Points

In this step, we will calculate the class wise MRL seperately for each component of the MTD in our dataset D.For calculating MRL, it uses a probalistic classifier GP. In order to achieve earliness, full length of MTD must not be used to predict class. The suggested approach first creates a dataset D_i by seperating i^{th} component from all the MTD.Then this dataset D_i is used to determine MRL for ith component. The GP classifier calculates the class probabilities for the time series C_i^j of the dataset D_i. For each component, minimum value of t is calculated such that we are able to acheive the desired accuracy α.

We can calculate class wise minimum required data points for a dataset following the given steps.

1. Build the GP classifier for the complete training dataset using all the data points.
2. Build the GP classifier for the complete training dataset using only f data points of each MTD.
3. Check for each class that if we are getting at least α accuracy. If yes, f is the required MRL for that class.
4. Increment f and repeat the process until we get the MRLs for all the labels.

2.3.2 Construct Classifiers

In this step we are going to use our estimated MRLs to build a set of classifiers. The classifier h_i is built with the help of the calculated MRD for the dataset D_i. Basically our main objective is to build a classifier that can classify an MTD as early as possible with at least α accuracy.In order to ensure α accuracy.

Now we will build GP classifiers for each sensor class wise using our MRLs. That means we will have a total of $n * l$ classifiers.

ALGORITHM 1: Construction of the classifiers

Input: Dataset D consists of N labeled MTD with n time series and l class labels, the lengt
T and λ_i, where $1 \leq i \leq n$;

Output: A set of classifiers $\mathbf{h} = \{h_1, h_2, \ldots, h_n\}$;

/* Construct classifier h_1 using full length of time series in dataset $\mathbf{D}_1$, using 1-NN*/

for *component* $i \leftarrow 2$ *to* n **do**
  for *label* $k \leftarrow 1$ *to* l **do**
    /* N_k is total number time series belonging to L_k */
    for $j \leftarrow 1$ *to* N_k **do**
      /* L_k is the label of $C_i^j \in \mathbf{D}$.
      /* Compute class posterior probabilities of C_i^j using GP by Equation (8).*/
      $\mathrm{p_{i,j,T}} = \{p_{(i,j,T,1)}, p_{(i,j,T,2)}, \ldots, p_{(i,j,T,l)}\}$.
      $\delta_T = max_1\{\mathrm{p_{i,j,T}}\} - max_2\{\mathrm{p_{i,j,T}}\}$.
      for $t = 1$ *to* T **do**
        $\mathrm{p_{i,j,t}} = \{p_{(i,j,t,1)}, p_{(i,j,t,2)}, \ldots, p_{(i,j,t,l)}\}$.
        $\delta_t = max_1\{\mathrm{p_{i,j,t}}\} - max_2\{\mathrm{p_{i,j,t}}\}$.
        if $(\delta_T \leq \delta_t)$ *and* $(\boldsymbol{\alpha} \times max_1\{\mathrm{p_{i,j,T}}\} \leq max_1\{\mathrm{p_{i,j,t}}\})$ **then**
          $\mathrm{MRD}(C_i^j) = t$.
    Obtain the natural group $\chi_{(i,L_k)}$ using Agglomerative hierarchical clustering.
    Find $MRD_{i,k}$ as the maximum of the MRD of the time series of G_{i,L_k}.
/* The classifier for ith component */
$h_i = \{MRD_{i,1}, MRD_{i,2}, \ldots, MRD_{i,l}\}$.
/* The classifiers for the dataset D */
$\mathbf{h} = \{h_1, h_2, \ldots, h_n\}$.
return h.

Figure 2.2 Calulating MRLs

2.3.3 Preparing transition matrix

For implementing HMM we need two things -

- Transition Matrix
- Observation Matrix

We will need to prepare transition matrix now and observation matrix at the time when we are in the process of predicting a class.

$$\begin{bmatrix} P_{11} & P_{12} & . & . & P_{1l} \\ P_{21} & P_{22} & . & . & P_{2l} \\ . & . & . & . & . \\ . & . & . & . & . \\ P_{l1} & P_{l2} & . & . & P_{ll} \end{bmatrix}$$

Figure 2.3 Representation of a transition matrix

Here P_{ij} denotes the probability of the next state being j if current state is i.

We will calculate these by treating the predicted class (from the GP classifiers) of a component as a state. Only the training dataset should be considered for the calucation of transition matrix.

2.3.4 Predicting the class of an incoming MTD

The following steps should be followed for each MTD.

Preparing observation matrix

Here we will use the class wise GP classifiers we built for each sensor/component. First we will choose the minimum MRL from our precalculated MRLs of the first component.

The predicted probabilities of a component can serve as a row of the observation matrix. The MRL used to predict probabilities for the next component should correspond to the class predicted for the previous component.

The maximum value of the MRL of a component should be considered to calculate percentage of data points used because we had to wait for those data points anyway.

Predicting the label using viterbi algorithm

Now we will predict the most probable sequence of the class labels using viterbi algorithm. That means the length of our sequence would be n.
The label that appears the most number of times would be our predicted label.

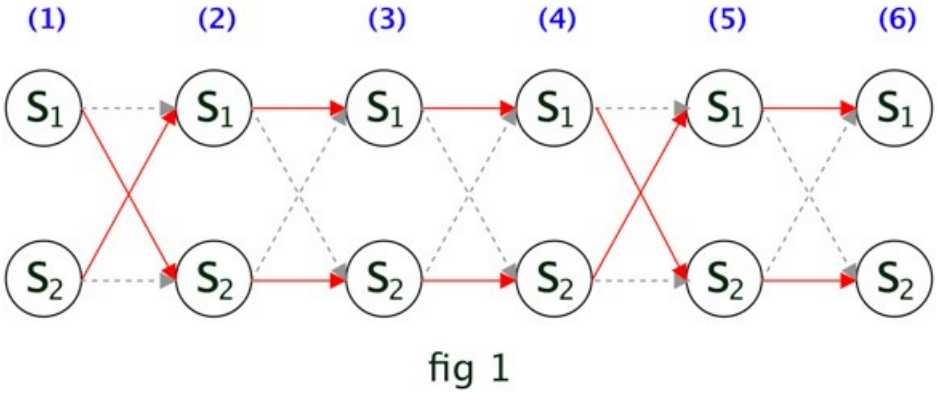

Figure 2.4 Representation of viterbi algorithm.

2.3.5 My implementation

Here is the link my implementation of this approach - https://github.com/nishantwrp/exploratory-project

Chapter 3

Conclusions and Discussion

In this report, we have discussed a HMM based classification algorithm for early classification of a multivariate time series with different sampling rates while maintaining the α accuracy. Our approach first calculate class-wise Minimum Required Data Points(MRL) using a probalistic GP classifier.Then afterwards, it construct a set of classifier using the calculate MRLs. Finally we use viterbi algorithm to find state sequence in our Hidden Markov Model.

Our accuracy and earliness of the experiments performed on PEMS-SF dataset illustrates that our approach outperforms the existing approaches of time series prediction. The proposed approach can be further extended to achieve better earliness using deep-learning models by optimizing the correlation among the components of MTD. Further, as the proposed approach is heavily dependent on the user given desired accuracy, a method can be developed to determine such accuracy based on application domain. Such a method will help to improve the effectiveness of the approach.

```
SUMMARY
-------------------  ------------------------------------------------------------------------
Algorithms Used      Gaussian Process Classifier, Hidden Markov Model using viterbi algorithm
Accuracy             95.45%
Data Used            85.48%
Max data points used  134
Avg data points used  123.09
Sensors              10
Training fraction    0.8
Alpha                0.9
-------------------  ------------------------------------------------------------------------

CONFUSION MATRIX
--  --  --  --  --  --  --
10   0   0   0   0   0   0
 0  10   0   0   0   0   0
 0   2  12   0   0   0   0
 0   0   0  12   0   0   0
 0   0   1   1  10   0   0
 0   0   0   0   0  13   0
 0   0   0   0   0   0  17
--  --  --  --  --  --  --

2 Wednesday(s) confused for Tuesday(s)
1 Friday(s) confused for Wednesday(s)
1 Friday(s) confused for Thursday(s)
```

Figure 3.1 Output of my implementation of this approach on the PEMS-SF dataset

Chapter 4

Bibliography

1. W. Pei, H. Dibeklioglu, D. M. J. Tax and L. van der Maaten, "Multivariate time-series classification using the hidden-unit logistic model", IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 4, pp. 920-931, 2018.

2. K. Li, S. Li and Y. Fu, "Early Classification of Ongoing Observation", Proceedings of ICDM, pp. 310-319, 2014.

3. S. Li, K. Li and Y. Fu, "Early Recognition of 3D Human Actions", ACM Transactions on Multimedia Computing Communications and Applications, vol. 14, no. 1, pp. 20:1-20:21, 2018.

4. M. F. Ghalwash and Z. Obradovic, "Early classification of multivariate temporal observations by extraction of interpretable shapelets", BMC Bioinformatics, vol. 13, no. 1, pp. 195-206, 2012.

5. Y. F. Lin, H. H. Chen, V. S. Tseng and J. Pei, "Reliable early classification on multivariate time series with numerical and categorical attributes", Proceedings of PAKDD, pp. 199-211, 2015.

6. D. Dheeru and E. Karra Taniskidou, "UCI machine learning repository", 2019, [online].