Yamuna Devi S *et al*, International Journal of Computer Science and Mobile Applications,
Vol.4 Issue. 6, June- 2016, pg. 42-53 **ISSN: 2321-8363**
Impact Factor: 4.123

ENSEMBLE METHODS FOR TIME SERIES CLASSIFICATION

Yamuna Devi S[1], Dr. K. Sasikala Rani[2] *M.E.,Phd[2]*, Pavithra M[3], Priyanga M[4]

M.E. Student, Head of the Department, M.E Student, M.E Student

Hindusthan Institute of Technology, Coimbatore, Tamilnadu India

Yamuname11@gmail.com

Abstract

For more than a decade, time series similarity search has been given a great deal of attention by data mining researchers. As a result, many time series representations and distance measures have been proposed. The works on time series similarity search relies on shape-based similarity matching. Some of the algorithms are potential for short time series data and some of the data mining approaches are more powerful for long time series data. In existing scenario, the method is introduced named as Collective of Transformation-Based Ensembles method (COTE). It is mainly used for increasing the classification accuracy than preceding research. Another algorithm is named as Time series classification (TSC) which is used for transformation process which is based on comparative features. COTE contains classifiers constructed in the time, frequency, change, and shapelet transformation domains combined in alternative ensemble structures. However it has issue with transformation process and hence accuracy of the classification is reduced significantly. To avoid this issue we go for proposed scenario. In proposed system, we introduced the concept called as run length transformation to improve the classification accuracy higher than existing system. The run length algorithm is improved along with genetic approach to produce the optimal features. In this scenario, the measures are considered as similarity coefficient, likelihood ratio and dynamic time warping (DTW). Based on the modified k- nearest neighbor distance concept the speed is increased and classification accuracy is improved prominently. From the experimental result we can conclude that our proposed scenario yields better classification performance rather than existing scenario.

Keywords: Collective of Transformation-Based Ensembles method (COTE), Time series classification (TSC), dynamic time warping (DTW).

1. Introduction

Data mining is used to extract the important data from the large repositories. Data mining can be executed on data signified in quantitative, textual or multimedia forms. Data mining application could use several parameters to inspect the data. They contain the concepts such as association, sequence analysis, classification, clustering and forecasting. Association is defined as patterns where one event is connected to other event, like purchasing pen and bag. Path analysis or sequence is described as patterns where one event guides to other event, like the birth of a child and buying diapers. Classification is defined as the identification of new patterns such as coincidence among duct type purchases and plastic sheeting purchases. Clustering is described as determining and visually documenting groups of previously unknown facts, like geographic location and brand preferences. Forecasting is nothing but determining patterns from which one can make reasonable predictions regarding future activities. Data mining tools are used to forecast the future trends and behaviors.

In general, data mining is used in extensive series of profiling practices like marketing, surveillance, fraud detection and scientific discovery. A prominent concept is that constructing an extracting model is part of superior process which contains everything from defining the basic crisis that the representation will resolve, to arranging the model into a functioning environment.Data mining methods are the result of long research process and product development. This development start while business data was first saved on computers, continued with improvements in data access, and more recently, generated expertise's that permit users for navigating via their data in real time.

Classification is defined as the process of analysis and categorized into different classes. It is used to predict class labels based on the specified model. The time series classification is defined as constructing the classification model depends on labeled time series and utilizes this model to classify the label of unlabeled time series. The feature extraction methods applied and build features from the time series data initially, then use the classification methods such as support vector machine, k nearest neighbor, regression and decision tree to the feature set. A novel time series primitive called as time series shapelets which is used to predict most informative data [1].

Time series similarity is an important and great attention in data mining applications. Several time series representation and distance measures are produced. The preceding research was focused on the shape based similarity matching for low dimensional dataset [2]. For high dimensional dataset, the ensemble approaches and shapelets approaches are introduced. The several data mining concepts and techniques are utilized to evaluate the time series classification data.

In [3], AbdhullahMueen et.al (2011) discussed about logical-shapelets for time series classification. This scenario is notices the issue through introducing a new algorithm that discovers shapelets in a smaller amount of time than present methods via an order of magnitude.In [4], Lexiang Ye et.al (2011) suggested a novel method for time series shapelets.In this research, we introduced novel time series primitive, time series shapelets, which focused these restrictions.. This scenario can utilize the distance to the shapelet, than the distance to the nearest neighbor to categorize objects.

In [5], Jason Lines et.al (2012) presented a shapelet transform for time series classification. The problem of time series classification (TSC), the research consider any real-valued structured information a time series, provides a particular machine learning task as the organizing of variablesis frequently essential in discovering the optimal selective features.In [6], Mustafa GokceBaydogan (2013) discussed a bag of features structure to categorize time series.It presents afeature-based method which can deal warping though in a different way from DTW.

In [7], Houtao Deng (2013) presented a time series forest for classification and feature extraction. A tree-ensemble technique, named as time series forest (TSF), is introduced for time series categorization.In this research work, the method is used to illustrate that the entrancegain increases the accuracy of TSF.In [8], Anthony Bagnall et.al discussed about run length transformation for discriminating among auto regressive time series. This research is handled the issue of classification datasets where each case is a long time series that could be precisely modeled through autoregressive series. The run length transform (RL) along with autoregressive parameters are used to reduce the time complexity and improve the classification accuracy.

2. Materials and Methods

2.1. Shapelet transformation for localized similarity

In this module, a shapelet transformation is performed for localized similarity in the given time series dataset. A good shapelet discriminates among classes by using shapelet distance. In this research, take the shapelet as S, length l and time series T and sDist is shapelet distance which is the minimum Euclidian distance between shapelet and any length l subseries of T. Let the set of length l subseries of T and symbolized as W_l then

sDist (S, T) = $min_{w \in W_l}(dist(s, w))$ (1)

In this scenario, good shapelet have a minimum sDist to sample features of one class and large class of sample features of any other class. We have to transform the original data by using the best shapelets as features where attribute i in instance j of the transformed data is sDist (S_i, T_j),where S_i is the best shapelet and T_j is the jth instance of original data. The given below algorithm is used to determine the shapelets and transformation of data.

Algorithm 1:

Shapelet cached selection (T, min, max, k)

kShapelets←∅

for all T_i in T do

Shapelets←∅

forl← min to max do

$W_{i,l}$←enerateCandidates(T_i, l)

for all subsequence S in $W_{i,l}$ do

D_S←findDistance(S, T)

Quality ←assessCandidate (S,D_S)

Shapelets.add(S, quality)

sortByQulaity (shapelets)

removeSelfSimilar(shapelets)

kShapelets←merge(k,kShapelets, shape lets)

returnkShapelets

It builds a single pass via original data, considering each subseries is as a shapelet candidate. The set of sDist values for every candidate is discovered by using find Distance and accessed by the f-state quality measure in the assess Candidate procedure. The best k shape lets are returned, after removing overlapping candidates in the method removeSelfSimilar. This scenario employ the length inference process to establish the suitable values to use as the minimum and maximum shapelet lengths, and produce a maximum of shapeless, where n is the size of the training set of the original data.

2.2. Frequency domain: period gram transformation

For original valued time series T=<□ $_1$, □ $_2$ ….. □ □>the discrete Fourier transform (DFT) represents T as a linear combination of sinusoidal functions with amplitudes a, b and phase □,

$$\square_{\square} = \sum_{\square=1}^{\square} \square_{\square} \cos(2\square.\square_{\square}.\square) + \square_{\square}\sin(2\square.\square_{\square}.\square)) \qquad (2)$$

The period gram is the series

P = <$\square_1, \square_2, \ldots\ldots\ldots \square_{\square}$>

Where

$$\square_{\square} = \sqrt{\square_{\square}^2 + \square_{\square}^2}$$

The period gram is the Fourier transform of the ACF. The spectrum and ACF are different characterizations of the same information. The ACF is more useful for finding low order dependencies between the terms; the period gram is more useful for detecting lower-frequency correlations than the ACF. The first DFT coefficient of a series with zero mean will be zero. Since we always work with normalized series, we can ignore this term.

2.3.Autocorrelation based transformation

The Autocorrelation function measures the interdependence of terms in the time domain, and is commonly used in statistics and speech processing to model data where there is a dependency between observations over a short period of time.Positive autocorrelation in a series generally indicates some form of persistence, in that the series tends to remain in the previous state, whereas negative autocorrelation is indicative of high volatility.The ACF of time series T is $\rho = < \rho_1, \rho_2, \ldots\ldots \rho_{m-1} >$ where l is the maximum lag, and the representation is as follows

$$\rho_k = \frac{E[(t_i - \mu_i).(t_{i+k} - \mu_{i+k})]}{\sigma_i.\sigma_{i+k}} \qquad (3)$$

$\rho_k$is evaluated from the data by r_k typically, where

$$r_k = \frac{\sum_{i=1}^{m-k}(t_i - \bar{t})(t_{i+k} - \bar{t})}{\sum_{i=1}^{m}(t_i - \bar{t})^2} \qquad (4)$$

The quantity r_k is the auto correlation coefficient at lag k and has range [-1, 1]. If the series T has been normalized to zero mean and unit variance, the calculation of r_k simplifies to

$$r_k = \sum_{i=1}^{m-k}(t_i . t_{i+k}) \qquad (5)$$

The auto correlation function is frequently used to fit an autoregressive model to time series. An AR model is of the form

$$t_i = c + \sum_{j=1}^{p} \emptyset_i t_{t-j} + \varepsilon_i \qquad (6)$$

Where c is a constant $\emptyset_i$ are model parameters and ε_i are random variables. It estimates the parameters $\emptyset_i$are found through evaluating the partial autocorrelation function.

2.4. Run length transformation

In this module, we have to perform the run length transformation to improve the accuracy of classification results than existing results. The number of runs in a series is commonly used as a test of independence. The run length distribution is a sequence of counts of the number occurrences of runs of length (1, 2, . . . , *s*) for some predefined *s*. To check the unequal length, the run length distribution is transformed to run length proportion through dividing the run sum value. Then, for classification problems, it is used to find the run length similarity which increases the accuracy results and transformation speed.

Run length transform algorithm

Input: a time series x of length m, the series mean μ and the max run length s

Output: a run length histogram r.

Initialize r to all zeroes

l=1

fori=2 to m do

if$(x_{i-1} < \mu \wedge x_i < \mu) \vee (x_{i-1} > \mu \wedge x_i > \mu)$ then

l=l+1;

else

if l<s then

rl=rl+1;

else

r_s=r_s + 1

End if

l=1;

end if

end for

return r

2.5. Heterogeneous ensemble

In this module, the heterogeneous ensemble approach is introduced for specified datasets in terms of frequency, change and shapelet transformation domains. The classification algorithms are such as k Nearest Neighbor (where k is set through cross validation), Naive Bayes, C4.5 decision tree, support vector machines and random forest as well as Bayesian network. Every classifier is allocated a weight based on the cross validation training accuracy and new information is categorized along with a weighted vote. The group of classifiers is selected to balance easy and composite that employs probabilistic, tree based and kernel based models. With the exception of k-NN, we do not optimize parameter settings for these classifiers via cross validation.

2.6. Elastic ensemble

For the time domain, we used elastic ensemble classifier. The elastic ensemble is a combination of nearest neighbor classifiers which utilizes the elastic distance measures.The dynamic time warping with warping set through cross validation is typically used benchmark.There have been a number of variants of DTW.These include a weighted version of DTW (WDTW) that replaces the warping window with a weight function to penalize against large warpings. The elastic measure encloses a distance measure depends on the Longest Common Subsequence (LCSS) problem, Edit Distance with Real Penalty (ERP), Time Warp Edit (TWE) distance and Move-Split-Merge (MSE).This module demonstrate that by combining the predictions of 1-NN classifiers built with these distance measures and using a voting scheme that weights according to cross-validation training set accuracy, we can significantly outperform DTWCV.

Algorithm procedure

1. Start
2. Initialize the parameters
3. Generate simple nearest neighbor classification
4. Use dynamic time warping with warping set through cross validation
5. Add weight version of dynamic time warping
6. Replace warping window with a weight function
7. Compute longest common subsequence, edit distance with real penalty, time warp edit distance and move split merge
8. Obtain elastic ensemble
9. End

2.7. COTE classification

In this module, the COTE classification method is used to expand the accuracy for the specified time series classification dataset. An excellent technique is included all possible classifiers in one ensemble. The flat combined of transform based ensemble weights the vote of every classifier through its cross validation accuracy on the training data. The mean position of flat-COTE is considerably superior to all of the other classifiers. The data is presented through the shapelet transform and, to a smallerdegree, the power spectrum and change domains, gives discriminatory features that are hard, if not impossible, to discover in the time domain.COTE is used to provide better performance for all specified dataset than other datasets.

2.8. Modified KNN classification method

In this module, the modified kNN classification approach is used to perform the time series classification. The main objective of this technique is allocating the class label of data based on k validated data attributes of the training dataset. And the validity of the data attributes in the training dataset is calculated. The modified kNN is implemented on any test dataset attributes.

Start

For x=1 to train size

Validity (x) = calculate validity of attributes

End for

Output label=weighted_MkNN (validity, test sample)

Return output label

End

Output: MkNN (training dataset and testing dataset)

3. Results and Discussion

In this section, the performance metrics are evaluated by using existing and proposed methodologies. The performance metrics are such as accuracy, precision, recall and f-measure. The existing COTE classification method is shown the lower performance in given time series dataset. The proposed run length transformation with genetic algorithm is shown the higher performance in given time series dataset. The proposed modified k nearest neighbor classification used to increase the accuracy of time series performance. From the experimental result, we can conclude that the proposed system is better than the existing system in terms of higher performance. An experimental result shows that the proposed method achieves superior performance.

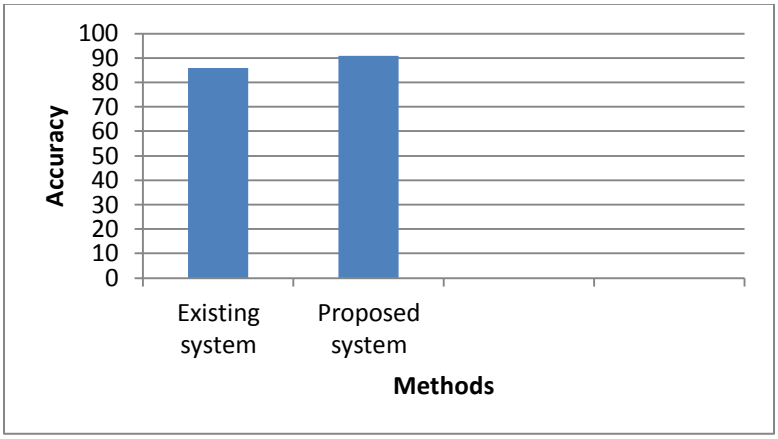
3.1. Accuracy

The accuracy is the proportion of true results (both true positives and true negatives) among the total number of cases examined.

Accuracycan be calculated from formula given as follows

$$\text{Accuracy} = \frac{\text{True positive} + \text{True negative}}{\text{True positive} + \text{True negative} + \text{False positive} + \text{False negative}}$$

An accuracy of 100% means that the measured values are exactly the same as the given values.



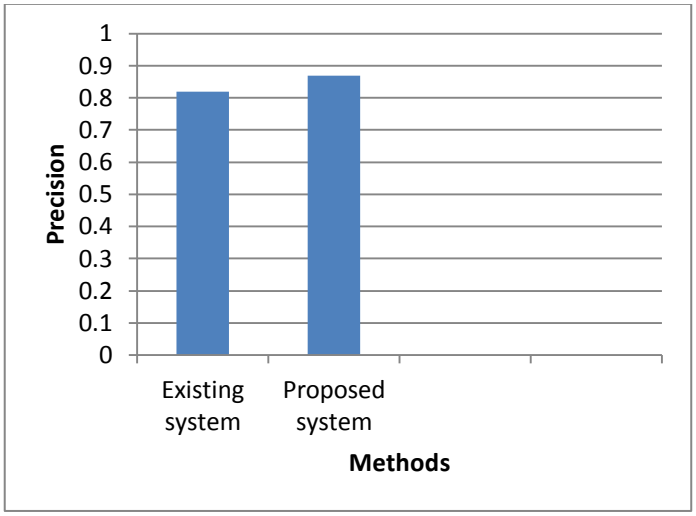
From the above graph we can observe that the comparison of existing and proposed system in terms of accuracy metric. In x axis we plot the methods and in y axis we plot the accuracy values. The accuracy values are lower by using existing algorithm. The accuracy value of existing scenario is 86. The accuracy value is higher by using the proposed method. The accuracy value of proposed scenario is 91. From the result, we conclude that proposed system is superior in performance.

3.2. Precision

The precision is calculated as follows:

$$\text{Precision} = \frac{\text{True positive}}{\text{True positive} + \text{False positive}}$$

Precision can be seen as a measure of exactness or quality, whereas recall is a measure of completeness or quantity. In simple terms, high precision means that an algorithm returned substantially more relevant results than irrelevant. In a classification task, the precision for a class is the number of true positives (i.e. the number of items correctly labeled as belonging to the positive class) divided by the total number of elements labeled as belonging to the positive class (i.e. the sum of true positives and false positives, which are items incorrectly labeled as belonging to the class).



From the above graph we can observe that the comparison of existing and proposed system in terms of precision metric. In x axis we plot the methods and in y axis we plot the precision values. The precision values are lower by using existing algorithm. The precision value of existing scenario is 0.82. The precision value is higher by using the proposed method. The precision value of proposed scenario is 0.87. From the result, we conclude that proposed system is superior in performance.

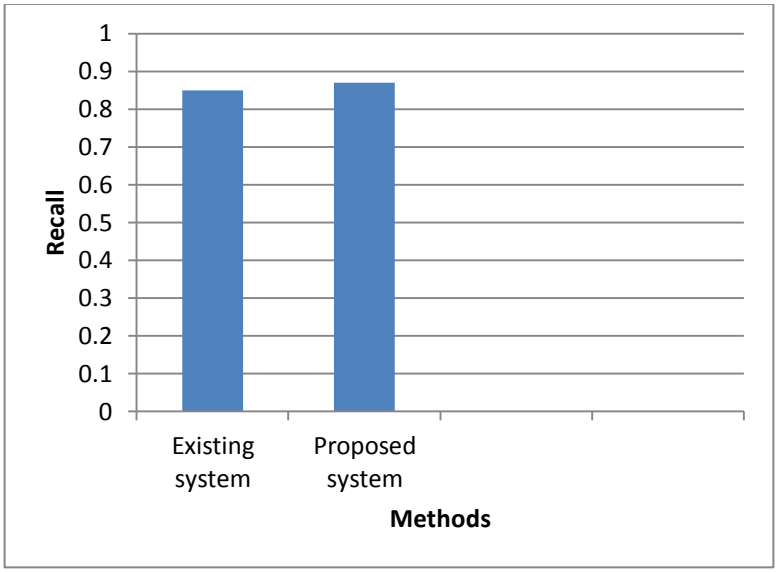
Recall

The calculation of the recall value is done as follows:

$$\text{Recall} = \frac{\text{True positive}}{\text{True positive} + \text{False negative}}$$

The comparison graph is depicted as follows:

Recall is defined as the number of relevant documents retrieved by a search divided by the total number of existing relevant documents, while precision is defined as the number of relevant documents retrieved by a search divided by the total number of documents retrieved by that search. Recall in this context is defined as the number of true positives divided by the total number of elements that actually belong to the positive class (i.e. the sum of true positives and false negatives, which are items which were not labeled as belonging to the positive class but should have been).

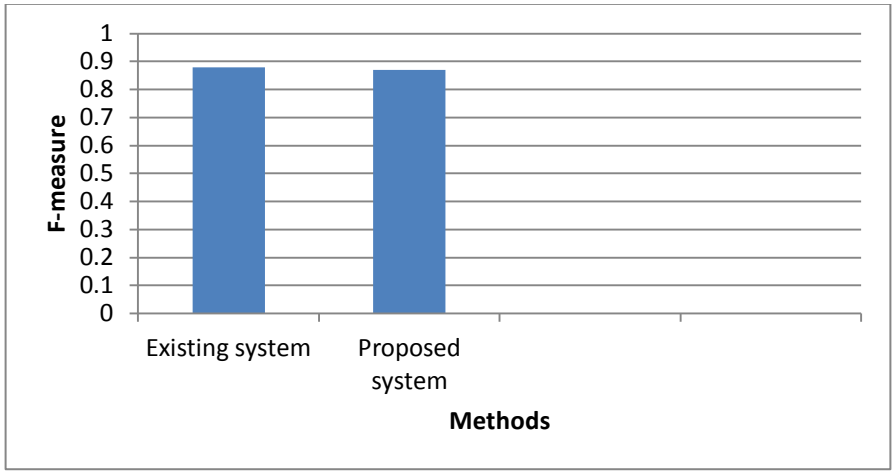


From the above graph we can observe that the comparison of existing and proposed system in terms of recall metric. In x axis we plot the methods and in y axis we plot the recall values. The recall values are lower by using existing algorithm. The recall value of existing scenario is 0.85. The recall value is higher by using the proposed method. The recall value of proposed scenario is 0.89. From the result, we conclude that proposed system is superior in performance.

F-measure

A measure that combines precision and recall is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score:

$$F = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}$$



From the above graph we can observe that the comparison of existing and proposed system in terms of f-measure metric. In x axis we plot the methods and in y axis we plot the f-measure values. The f-measure values are lower by using existing algorithm. The f-measure value of existing scenario is 0.88. The f-measure value is higher by using the proposed method. The f-measure value of proposed scenario is 0.91. From the result, we conclude that proposed system is superior in performance.

4. Conclusion

The existing system of COTE classification algorithm is used in TSC research such as 1-NN along with Euclidean distance and/or dynamic time warping. The COTE algorithm is considerably performs also interaction among classifier and transformation is more restrained. The existing system is focused on the time series classification problems and improved the accuracy results. In the proposed system, the run length transformation with genetic algorithm is used to increase the accuracy higher. It reduces the time complexity as well as increase the classification accuracy in superior by using MkNN classification approach.From the experimental result we can conclude that, proposed scenario yields higher performance rather than existing scenario. The performance is superior in terms of precision, recall, f-measure and accuracy values. Hence the proposed method is higher accuracy rather than existing scenario.

References

1. Ye, Lexiang, and Eamonn Keogh, Time series shapelets: a new primitive for data mining, *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2009.

2. Lin, Jessica, Rohan Khade, and Yuan Li, Rotation-invariant similarity in time series using bag-of-patterns representation, *Journal of Intelligent Information Systems* 39.2 (2012): 287-315.

3. Mueen, Abdullah, Eamonn Keogh, and Neal Young, Logical-shapelets: an expressive primitive for time series classification, *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2011.

4. Ye, Lexiang, and Eamonn Keogh, Time series shapelets: a novel technique that allows accurate, interpretable and fast classification, *Data mining and knowledge discovery* 22.1-2 (2011): 149-182.

5. Lines, Jason, et al, A shapelet transform for time series classification, *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2012.

6. Baydogan, Mustafa Gokce, George Runger, and Eugene Tuv, A bag-of-features framework to classify time series, *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35.11 (2013): 2796-2802.

7. Deng, Houtao, et al, A time series forest for classification and feature extraction, *Information Sciences* 239 (2013): 142-153.

8. Bagnall, Anthony, and Gareth Janacek. "A Run Length Transformation for Discriminating Between Auto Regressive Time Series." *Journal of Classification* 31.2 (2014): 154-178.

