Time series classification based on multi-feature dectionary representation and ensemble learning

Anzellotti Alberto
Martinico Silvio
Pitzalis Nicola

University of Pisa
Data Mining

Time series classification

Preliminary definitions

- **Time series**: $T = (t_1, \ldots, t_n),\ t_i \in \mathbb{R}\ \forall\, i \in \{1, ..., n\}$
- **Subsequence of** T: $S = (t_i, \ldots, t_j)$ for $1 \le i \le j \le n$
- **Sliding window**: $G := \{S_1^w, \ldots S_{n-w+1}^w\}$ where $S_k^w := (t_k, \ldots, t_{k+w-1})$

We need to define similarity between time series in order to classify them

Classification algorithms

Two categories of algorithms:

- Whole-series-based: consider distance measure on the whole series (ED or DTW) and use 1-NN to make classification
- Feature-based: Transform the original time series into a set of feature vectors

Time series classification

Preliminary definitions

- **Time series**: $T = (t_1, \ldots, t_n),\ t_i \in \mathbb{R}\ \forall\, i \in \{1, ..., n\}$
- **Subsequence of** T: $S = (t_i, \ldots, t_j)$ for $1 \leq i \leq j \leq n$
- **Sliding window**: $G := \{S_1^w, \ldots S_{n-w+1}^w\}$ where $S_k^w := (t_k, \ldots, t_{k+w-1})$

We need to define similarity between time series in order to classify them

Classification algorithms

Two categories of algorithms:

- Whole-series-based: consider distance measure on the whole series (ED or DTW) and use 1-NN to make classification
- Feature-based: Transform the original time series into a set of feature vectors

Feature-based classification algorithms

Three categories:

- Intervals based
- Shapelets based
- Dictionary based

Dictionary-based algorithms

- Have recently reached promising classification accuracy
- Exploits repeated patterns basing on their frequency
- Classify time series basing on histograms

Two main types:

- **Symbolic Aggregate approXimation** (SAX)
- Symbolic Fourier Approximation (SFA)

Feature-based classification algorithms

Three categories:

- Intervals based
- Shapelets based
- Dictionary based

Dictionary-based algorithms

- Have recently reached promising classification accuracy
- Exploits repeated patterns basing on their frequency
- Classify time series basing on histograms

Two main types:

- **Symbolic Aggregate approXimation** (SAX)
- Symbolic Fourier Approximation (SFA)

Symbolic Aggregate approXimation (SAX)

- Proposed by Lin et al. in 2007
- Transforms time series into symbol sequences
- Reduces dimensionality
- Makes data mining tasks more efficient

However, SAX methods fail to reach state of the art accuracy

Major issues

1. Only extracts the mean feature
2. Uses one single classifier, sharing the same hyperparameters for all patterns

...this is not enough

Symbolic Aggregate approXimation (SAX)

- Proposed by Lin et al. in 2007
- Transforms time series into symbol sequences
- Reduces dimensionality
- Makes data mining tasks more efficient

However, SAX methods fail to reach state of the art accuracy

Major issues

1. Only extracts the mean feature
2. Uses one single classifier, sharing the same hyperparameters for all patterns

...this is not enough

Symbolic Aggregate approXimation (SAX)

- Proposed by Lin et al. in 2007
- Transforms time series into symbol sequences
- Reduces dimensionality
- Makes data mining tasks more efficient

However, SAX methods fail to reach state of the art accuracy

Major issues

1. Only extracts the mean feature
2. Uses one single classifier, sharing the same hyperparameters for all patterns

…this is not enough

SAX issues explained: Mean is not enough

1. Same segment mean, completely different trend

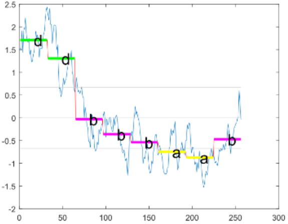


Fig. 2. An example of SAX representation.

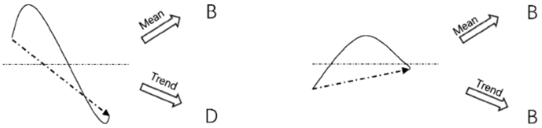


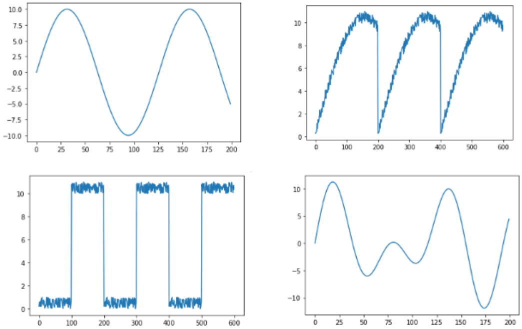
Fig. 1. Example of two segments with two feature symbols.

The "mean" feature is not enough to represent a T.S. for classification

SAX issues explained: Mean is not enough

1. Same segment mean, completely different trend

Different patterns are better detected by different classifiers. This problem could be solved ensembling different models with different confidence values



This paper contributions

1. Trend feature extraction

- Less information loss
- Still computationally viable

2. TBOP / TBOPE

- Trend + Bag Of Patterns
- Extension to Ensemble model with voting

3. UCR TS dataset benchmarking

- Time series datasets widely used as benchmark for time series classification
- Comparison with SFA-based ensemble classifiers, SAX-based shapelets method, interval algorithms and 1NN-DTW

This paper contributions

1. Trend feature extraction

- Less information loss
- Still computationally viable

2. TBOP / TBOPE

- Trend + Bag Of Patterns
- Extension to Ensemble model with voting

3. UCR TS dataset benchmarking

- Time series datasets widely used as benchmark for time series classification
- Comparison with SFA-based ensemble classifiers, SAX-based shapelets method, interval algorithms and 1NN-DTW

This paper contributions

1. Trend feature extraction

- Less information loss
- Still computationally viable

2. TBOP / TBOPE

- Trend + Bag Of Patterns
- Extension to Ensemble model with voting

3. UCR TS dataset benchmarking

- Time series datasets widely used as benchmark for time series classification
- Comparison with SFA-based ensemble classifiers, SAX-based shapelets method, interval algorithms and 1NN-DTW

Trend feature extraction

Key Points

- Considering also the trend is key to good classification

Trend feature extraction

Key Points

- Considering also the trend is key to good classification
- Methods for trend features such as PLA are computationally expensive

Trend feature extraction

Key Points

- Considering also the trend is key to good classification
- Methods for trend features such as PLA are computationally expensive
- No generally accepted criterion for time series trend evaluation

Trend feature extraction

Key Points

- Considering also the trend is key to good classification
- Methods for trend features such as PLA are computationally expensive
- No generally accepted criterion for time series trend evaluation
- Small segments often have (approximately) linear trends

Trend feature extraction

Key Points

- Considering also the trend is key to good classification
- Methods for trend features such as PLA are computationally expensive
- No generally accepted criterion for time series trend evaluation
- Small segments often have (approximately) linear trends
- Segment of length p:

$$s = \left[t_k, \ldots, t_{k+p-1}\right]$$

Trend feature extraction

Key Points

- Considering also the trend is key to good classification
- Methods for trend features such as PLA are computationally expensive
- No generally accepted criterion for time series trend evaluation
- Small segments often have (approximately) linear trends
- Segment of length p:

$$s = \left[t_k, \ldots, t_{k+p-1}\right]$$

- Definition of trend:

$$trend(s) := \frac{t_{k+p-1} - t_k}{p-1}$$

Observations

- Computing this trend is cheap

Observations

- Computing this trend is cheap
- Retains much more information compared to the mean alone

Observations

- Computing this trend is cheap
- Retains much more information compared to the mean alone
- SAX requires the trend to be further symbolized to meet the STD:
  - Let $dif(s) := \left[t_{k+1} - t_k, t_{k+2} - t_{k+1}, \ldots, t_{k+p-1} - t_{k+p-2}\right]$ be the series of time deltas, it is obvious that:

$$M(dif(s)) = \frac{(t_{k+1} - t_k) + (t_{k+2} - t_{k+1}) + \cdots + (t_{k+p-1} - t_{k+p-2})}{p-1} = \frac{t_{k+p-1} - t_k}{p-1}$$

  - In absence of normalization $M(dif(s)) = trend(s)$

Observations

- Computing this trend is cheap
- Retains much more information compared to the mean alone
- SAX requires the trend to be further symbolized to meet the STD:
  - Let $dif(s) := \left[t_{k+1} - t_k, t_{k+2} - t_{k+1}, \ldots, t_{k+p-1} - t_{k+p-2}\right]$ be the series of time deltas, it is obvious that:

$$M(dif(s)) = \frac{(t_{k+1} - t_k) + (t_{k+2} - t_{k+1}) + \cdots + (t_{k+p-1} - t_{k+p-2})}{p-1} = \frac{t_{k+p-1} - t_k}{p-1}$$

  - In absence of normalization $M(dif(s)) = trend(s)$
- Therefore we compute the trend as $SAX\big(dif(s)\big)$

TBOP / TBOPE

Time Series Similarity

- Usually defined as the ED between histograms
- Not possible with two histograms for each time series

$$HistSim(T_1, T_2) := \cos(H_m^1, H_m^2)\cos(H_t^1, H_t^2)$$

$$\cos(H_m^1, H_m^2) = \frac{H_m^1 \cdot H_m^2}{||H_m^1||\,||H_m^2||}$$

$$\cos(H_t^1, H_t^2) = \frac{H_t^1 \cdot H_t^2}{||H_t^1||\,||H_t^2||}$$

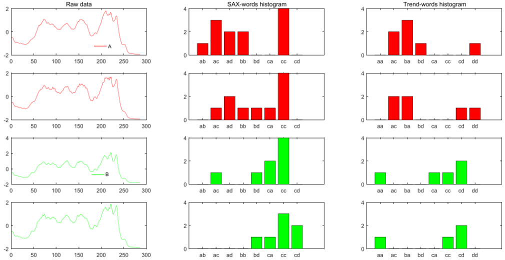


Fig. 4. An example of TBOP. (left) raw data. (middle) histogram of SAX words. (right) histogram of trend words.

Algorithm

Algorithm 2: TBOP classification

Input: training data $\mathscr{D}$, testing time series T
Output: classlabel

function classify T
    $bestSim \leftarrow 0, bestTS \leftarrow \varnothing, Sim \leftarrow 0$
    for each *Sample* in $\mathscr{D}$ **do**
        $Sim \leftarrow HistSim(T, Sample[i])$
        if $Sim > bestSim$ **then**
            $bestSim \leftarrow Sim$
            $bestTS \leftarrow Sample[i]$
        end if
    end for
    return $bestTS.classlabel$
end function

Ensemble

BOP - patterns extraction

- Input T and dif
- Computes hidden features
  - SAX(T) + SAX(dif(T))

For each model

- Input hidden features
- Computes similarity
- Extract class label

For the ensemble

- Compute models confidence
- Choose final class label

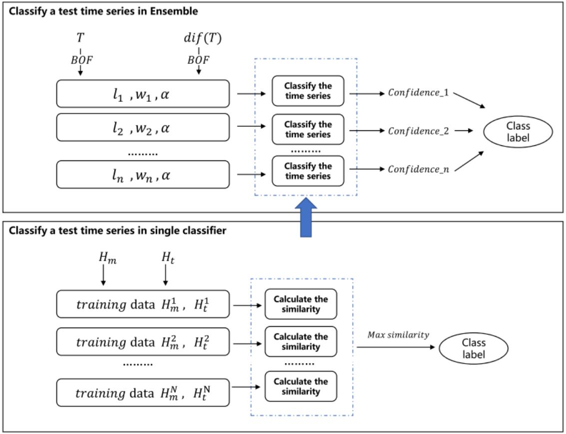


Fig. 3. Workflow of the TBOPE algorithm.

Ensemble

BOP - patterns extraction

- Input T and dif
- Computes hidden features

  SAX(T) + SAX(dif(T))

For each model

- Input hidden features
- Computes similarity
- Extract class label

For the ensemble

- Compute models confidence
- Choose final class label

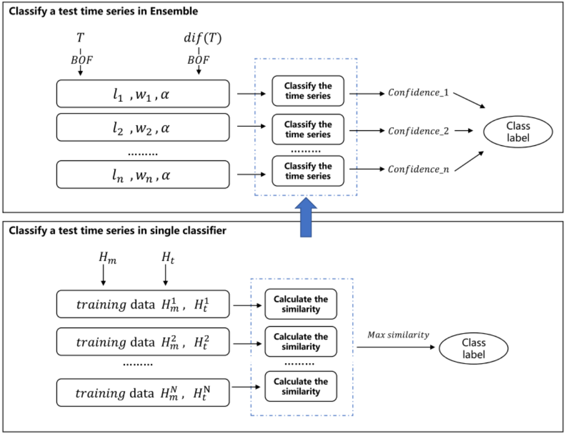


Fig. 3. Workflow of the TBOPE algorithm.

Ensemble

BOP - patterns extraction

- Input T and dif
- Computes hidden features
  SAX(T) + SAX(dif(T))

For each model

- Input hidden features
- Computes similarity
- Extract class label

For the ensemble

- Compute models confidence
- Choose final class label

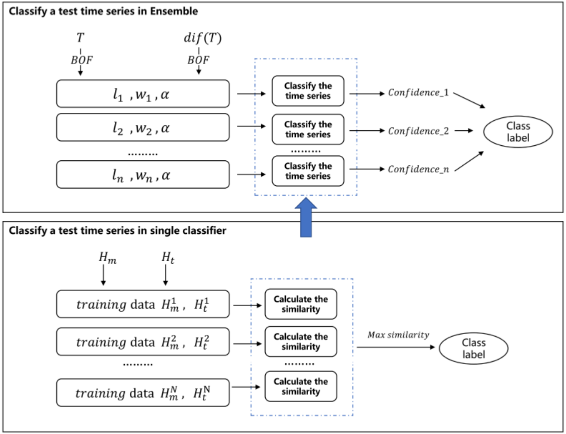


Fig. 3. Workflow of the TBOPE algorithm.

Ensemble choice

Algorithm 3: Select single classifiers for ensemble

```
Input: 𝒟 training data,minWin, maxWin, minLen, maxLen
Output: Ens
functionSelectClassifiers
    bestCVacc←0, ensemble Ens←∅, single classifier Sc←∅
    for w = minWin to maxWin do
      for l = minLen to maxLen do
        (CVacc, confidence)←CrossvalidationSc(D, w, l)
        if (CVacc > bestCVacc ∗ factor) then
          add Sc to Ens
          if (CVacc > bestCVacc||ensemble.size > maxsize) then
            update Ens
          end if
        end if
      end for
    end for
    return Ens
end function
```

Ensemble classification

Algorithm 4: TBOPE Ensemble classification

```
Input: unclassified time series T, ensemble Ens
Output: classlabel
function ensembleclassification
    classHist←∅
    for each classifier Sc in Ens do
      if i = Sc.Classify(T) then
        classHist[i] + = Sc.confidence[i]
      end if
    end for
    classlabel = max(classHist)
    return classlabel
end function
```

Experiments

UCR TS dataset benchmarking

- 82 datasets from UCR dataset, widely used as benchmark datasets for time series classification (Training set + Test set)
- TBOPE ensemble classifier implemented in JAVA

Setup

- Using 4 symbols performs best
- $factor = 0.92$ to ensure we select many classifiers
- Ensemble size not limited
- $maxLen = \frac{L}{2}$ so that trend approximation is reliable

Table 1
Parameter setting.

Parameter	Value
l	{2,3,4,5,6,7,8}
α	4
w	$maxLen = L/2, minLen = L/50$
$factor$	0.92

Experiments

UCR TS dataset benchmarking

- 82 datasets from UCR dataset, widely used as benchmark datasets for time series classification (Training set + Test set)
- TBOPE ensemble classifier implemented in JAVA

Setup

- Using 4 symbols performs best
- $\mathit{factor} = 0.92$ to ensure we select many classifiers
- Ensemble size not limited
- $\mathit{maxLen} = \frac{L}{2}$ so that trend approximation is reliable

Table 1
Parameter setting.

Parameter	Value
l	{2,3,4,5,6,7,8}
α	4
w	$\mathit{maxLen} = L/2, \mathit{minLen} = L/50$
factor	0.92

Training time

Despite it also extract trend features, TBOPE is still slightly better than BOP.

TBOPE adopts the strategy of early abandonment to avoid some invalid calculations during the cross-validation process.

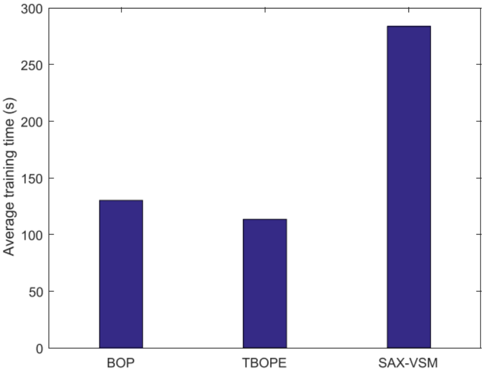


Fig. 6. Average training time for different classifiers.

Comparisons

Accuracy

- No significant difference between TBOPE and BOSS
- SFA is more sensitive to parameters so that it is better at extracting valid feature under different parameter combinations
- SFA is harder to interpret

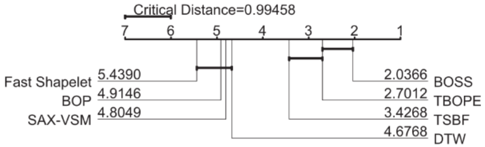


Fig. 5. Critical difference diagram of average ranks on accuracy for seven classifiers.

Choice of factor

- Factor cannot be too small for filtering classifiers, so we tested for values > 0.85

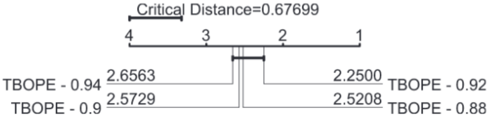


Fig. 7. Critical difference diagram of average ranks on accuracy for different factors.

Influence of design decisions

Trend VS No Trend

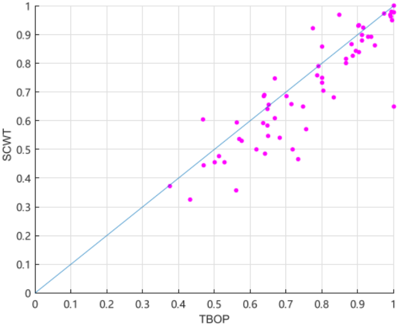


Fig. 8. SCWT vs TBOP.

Influence of design decisions

Trend VS No Trend

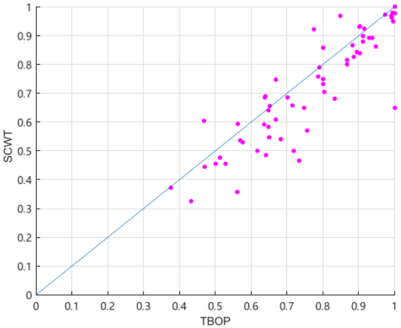


Fig. 8. SCWT vs TBOP.

Ensemble VS No Ensemble

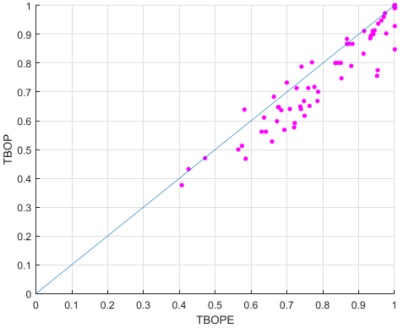


Fig. 9. TBOPE vs TBOP.

Grazie per l’attenzione!