Omid Abbaszadeh, Ali Amiri, Ali Reza Khanteymoori, 2015. An ensemble method for data stream classification in the presence of concept drift. *Frontiers of Information Technology & Electronic Engineering*, **16**(12):1059-1068. [doi:10.1631/FITEE.1400398]

An ensemble method for data stream classification in the presence of concept drift

Key words: Data stream, Classification, Ensemble classifiers, Concept drift

Contact: Omid Abbaszadeh
E-mail: o.abbaszadeh@znu.ac.ir
ORCID: http://orcid.org/0000-0002-8923-940X

Introduction

- ‘Data stream’ refers to a continuous and infinite series of data with a high generation rate.
- An issue for the data stream is classification of input data.
- Specific features of data streams are immense volume, high production rate, limited data processing time, and data concept drift; these features differentiate the data stream from standard types of data.
- A novel ensemble classifier is proposed in this paper.

Proposed method

Any ensemble classifier for processing data stream should be able to show:

- How the class label of an input data will be predicted?
  - ➢ We used new dynamic weighting mechanisms under different data input conditions.
- How concept drift is detected?
  - ➢ We used Kappa statistics for concept drift detection.
- How an expert is dropped from the set of classifiers?
  - ➢ We used new quality measures for dropping the poor classifiers.

Weighting mechanism

Algorithm 1 Weighting

Require: S_i: data stream; C_i: example chunk of size c; E: ensemble

Ensure: E_w: weighted ensemble E

if size$(C_i) = c$ && drift is detected **then**
    for each e in E **do**
        $e_\mathrm{weight} = W_\mathrm{Linear}$
        Add e to E_w
    end for
end if
if size$(C_i) = c$ && drift is not detected **then**
    for each e in E **do**
        $e_\mathrm{weight} = W_\mathrm{Non\_linear}$
        Add e to E_w
    end for
end if
return E_w

Drift detection

Algorithm 2 Drift detection

Require: C_{100}: last chunk of size 100; E: ensemble; α: agreement factor

Ensure: True, False

//Default value ($\alpha = 65\%$)
if (C_{100}) && $\kappa(E) > \alpha$ **then**
    return False
end if
if (C_{100}) && $\kappa(E) \leq \alpha$ **then**
    return True
end if

Expert(s) deletion

Algorithm 3 Expert(s) deletion

Require: S_i: data stream; C_i: example chunk of size c; E: ensemble

Ensure: $\text{poorest}_{\text{classifiers}}$

$\alpha_{\min} = 1.0$
$\text{poorest}_{\text{classifiers}}$=null
if size(C_i) $= c$ && drift is not detected **then**
  if size(E) $=$ default ensemble size **then**
    for each e in E **do**
      if $e_\alpha \leq \alpha_{\min}$ **then**
        $\alpha_{\min} = e_\alpha$
        $\text{poor}_{\text{classifier}} = e$
      end if
    end for
    $\text{poorest}_{\text{classifiers}}$.add($\text{poor}_{\text{classifier}}$)
  end if
end if
if size(C_i) $= c$ && drift is detected **then**
  for each e in E **do**
    $\alpha_{\text{total}} += 1/(e_\alpha + \epsilon)$
  end for
  $\alpha_{\text{avg}} = |E|/\alpha_{\text{total}}$
  for each e in E **do**
    if $e_\alpha < \alpha_{\text{avg}}$ **then**
      $\text{poorest}_{\text{classifiers}}$.add($e$)
    end if
  end for
end if
return $\text{poorest}_{\text{classifiers}}$

Experimental result

Table 3 Average classification accuracies (%)

Dataset	Proposed	DWM	OZA	CVFDT	AWE
SEA_S	84.84	84.82	82.64	87.71	87.19
SEA_G	87.37	84.91	83.37	85.00	85.10
$Hyper_S$	85.70	88.70	71.65	82.40	87.30
$Hyper_M$	79.90	76.84	71.79	71.36	72.17
LED_M	74.10	73.95	71.63	68.11	73.58
Wave	85.71	83.82	83.37	83.90	81.57
$Wave_M$	84.15	83.75	83.22	82.71	81.31

S: sudden drift; G: gradual drift; M: combination of drifts including sudden and gradual drifts

Experimental result (Con'd)

Table 5 Average memory usage

Dataset	Memory (MB)			
	Proposed	DWM	OZA	AWE
SEA_S	1.56	1.32	6.23	2.66
SEA_G	1.22	1.73	4.82	1.93
$Hyper_S$	3.38	4.24	11.31	3.41
$Hyper_M$	3.22	4.37	10.19	3.71
LED_M	0.48	0.61	2.56	0.32
Wave	56.30	6.18	69.73	50.63
$Wave_M$	15.81	6.42	26.16	12.29

S: sudden drift; G: gradual drift; M: combination of drifts including sudden and gradual drifts

Experimental result (Con'd)

Table 4 Average classification accuracy and time using different window sizes

Window size	Time (s)					Accuracy (%)				
	SEA_S	SEA_G	$Hyper_S$	$Hyper_M$	Wave	SEA_S	SEA_G	$Hyper_S$	$Hyper_M$	Wave
500	40.19	38.72	41.95	106	2120.96	84.84	87.37	85.70	79.90	85.71
750	39.83	36.70	41.25	104	2110.91	84.57	87.14	85.50	78.84	85.94
1000	37.92	36.45	40.06	109	2111.87	84.21	86.79	85.29	78.80	86.29
1250	37.45	35.70	39.84	103	2106.12	83.94	86.52	85.47	78.51	86.52
1500	36.30	35.12	38.88	99	2097.11	83.92	86.25	85.23	79.33	86.50
1750	35.59	34.41	38.02	98	2081.21	83.68	85.95	85.62	78.91	86.41
2000	34.88	33.58	37.05	96	2064.91	83.30	85.64	85.12	78.11	86.83

S: sudden drift; G: gradual drift; M: combination of drifts including sudden and gradual drifts

Experimental result (Con'd)

Table 6 Average classification precision

Dataset	Classification precision (%)			
	Proposed	DWM	OZA	AWE
SEA_S	66.56	65.59	60.92	69.96
SEA_G	71.62	66.15	62.70	68.72
$Hyper_S$	69.50	76.82	41.58	74.59
$Hyper_M$	55.41	53.11	40.81	43.84
LED_M	71.20	70.33	67.78	69.92
Wave	79.10	75.63	74.95	72.26
$Wave_M$	80.19	73.86	74.57	71.70

S: sudden drift; G: gradual drift; M: combination of drifts including sudden and gradual drifts

Experimental result (Con'd)

Table 7 Average of time consumption for 1000 test examples

Dataset	Time (s)			
	Proposed	DWM	OZA	AWE
SEA_S	0.19	0.07	7.94	0.14
SEA_G	0.09	0.06	5.97	0.09
$Hyper_S$	0.31	0.20	9.16	0.20
$Hyper_M$	0.24	0.21	3.90	0.22
LED_M	0.54	0.15	0.75	0.15
Wave	3.67	0.48	33.65	2.97
$Wave_M$	3.87	0.46	33.46	1.05

S: sudden drift; G: gradual drift; M: combination of drifts including sudden and gradual drifts

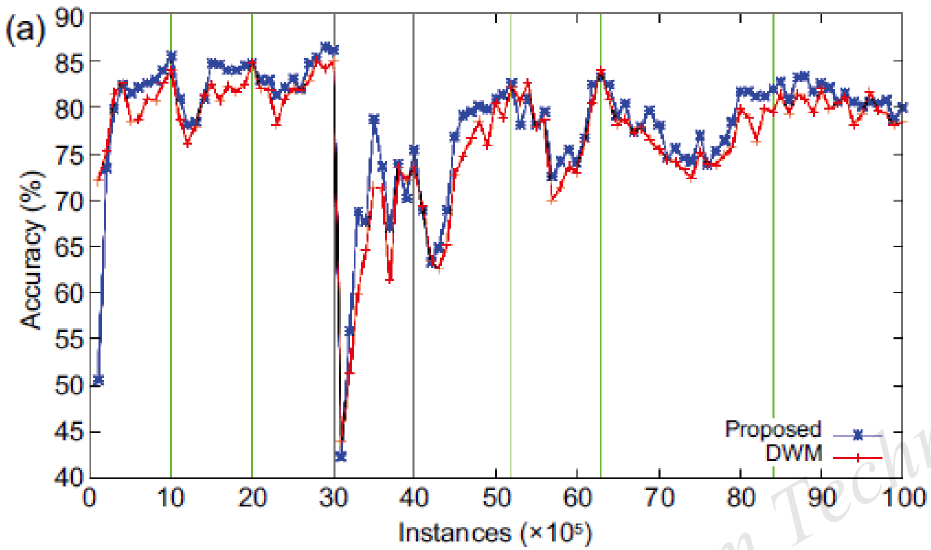
Experimental result (Con'd)

Table 8 Time consumption for training and testing in datastream

Dataset	Time (s)			
	Proposed	DWM	OZA	AWE
SEA_S	40.19	41.22	221.09	43.22
SEA_G	34.71	35.93	206.11	38.66
$Hyper_S$	41.95	47.08	327.40	48.73
$Hyper_M$	80.82	74.09	260.55	79.09
LED_M	430.09	384.77	597.00	369.56
Wave	2230.44	2111.20	18932.00	2120.96

S: sudden drift; G: gradual drift; M: combination of drifts including sudden and gradual drifts

Experimental result (Con'd)



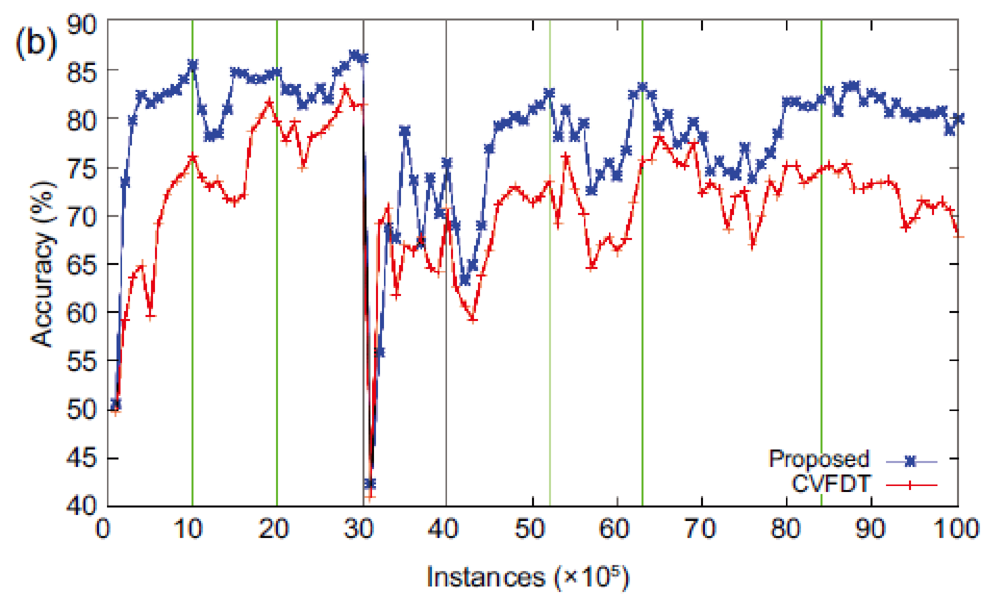


Fig. 1 Comparison of classification accuracy on the $\mathrm{Hyper}_{\mathrm{M}}$ dataset: (a) the proposed method vs. DWM; (b) the proposed method vs. CVFDT. The black and green horizontal lines represent sudden and gradual drifts, respectively. The first drift occurs at time 0.

Conclusions

- As the comparisons based on standard data sets show, the proposed method has acceptable accuracy under gradual and combined drifts.
- Compared with other methods, the proposed method approaches concept drift differently and uses a precision measure to detect drift.
- Good harmony is another advantage of the algorithm, which is better with gradual and mixed drift than with sudden drift.