# A Concept-Drifting Detection Algorithm for Categorical Evolving Data

Fuyuan Cao[1,2] and Joshua Zhexue Huang[1]

[1] Shenzhen Key Laboratory of High Performance Data Mining, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
[2] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, the School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China
cfy@sxu.edu.cn, zx.huang@siat.ac.cn

**Abstract.** In data streams analysis, detecting concept-drifting is a very important problem for real-time decision making. In this paper, we propose a new method for detecting concept drifts by measuring the difference of distributions between two concepts. The difference is defined by approximation accuracy of rough set theory, which can also be used to measure the change speed of concepts. We propose a concept-drifting detection algorithm and analyze its complexity. The experimental results on a real data set with a half million records have shown that the proposed algorithm is not only effective in discovering the changes of concepts but also efficient in processing large data sets.

**Keywords:** Categorical Data, Evolving, Concept-drifting.

## 1 Introduction

Many real world applications generate continuously arriving data, such as business transactions, web logs, sensors networks, etc. This type of data is known as data streams [1]. Generally speaking, a data stream can be considered as a sequence of items of structural information in which each item is stamped with a time point. As the arrival items change with time, the data distribution of the underlying structural information may change as well. Usually, the cause of the change is unknown. To understand the behaviors of data streams, it is important to investigate the changes of the distributions and the causes of the changes.

Semantically, the distribution of the structural information at a particular time point in a data stream is referred to as representation of a concept. A concept is defined by its intension and extension. Intension is the representation schema of structural information while extension refers to the set of objects represented by the schema. A concept often contains a set of sub-concepts. In machine learning, we can learn the intensions of concepts or sub-concepts from a set of objects. In supervised learning, every object is labeled with a class in the target variable. The set of objects in the same class is referred to as a sub-concept. In unsupervised learning, the classes of objects can be obtained with a clustering algorithm. In this case, a cluster is a sub-concept.

J. Pei et al. (Eds.): PAKDD 2013, Part II, LNAI 7819, pp. 485–496, 2013.

As the arrival items change over time, the change of data distribution can be used to induce the change of a concept. In real applications, the change of a concept is mainly caused by emerging new sub-concepts or fading old sub-concepts or both. A radical change of a concept is often known as concept drift [2]. Two kinds of concept drift are illustrated in literature [3]. One is sudden (abrupt) concept drift and the other is gradual concept drift. Sudden concept drift is described as that the data distribution is dramatically changed in a short time period. Gradual concept drift is considered that the change of a concept occurs gradually over time. For example, in social network analysis, different groups of people are interested in different topics. Some people may gradually change their interests from one topic to another over time and some may suddenly change their interests to new topics.

To investigate the behaviors of such data streams, we concern whether the concept at time $t_2$ has drifted from the concept at time $t_1$, where $t_2 > t_1$. Meantime, we are interested in the change speed of concepts.

In this paper, we propose a new method to measure the difference between two concepts at the different time points. This difference is defined by approximation accuracy of rough set theory. Based on the new measure, we propose a concept-drifting detection algorithm to detect whether a concept has drifted or not. We have conducted a series of experiments on the KDD-CUP'99 data. The experimental results have demonstrated the proposed algorithm is not only effective in discovering the changes of concepts but also efficient in processing large data sets.

## 2 Preliminaries

In this section, we first review the basic concepts in rough set theory [4], such as indiscernibility relation, lower and upper approximations, approximation accuracy that are used to define the measures of concept change. We then define the problem of concept-drifting in the categorical time-evolving data.

### 2.1 Some Basic Concepts of Rough Set Theory

In a relational database, the structural data is stored in a table, where each row represents an object and each column represents an attribute that describes the objects. Formally, a data table can be defined as a quadruple $DT = (U, A, V, f)$, where $U$ is a nonempty set of objects called the universe and $A$ is a nonempty set of attributes such that for any $x \in U$ and $a \in A$, $f(x, a) \in V_a$. $V = \bigcup_{a \in A} V_a$ is the union of all attribute domains. If $V$ is represented by continuous values, then $DT$ is called a numerical data table. For any $a \in A$, if $V_a$ is finite and unordered, then $DT$ is called a categorical data table. Unless otherwise specified, $DT$ represents a categorical data table in this paper.

Let $DT$ be a categorical data table defined on $A$ and $P \subseteq A$. $P$ defines an equivalence relation $IND(P)$ as:

$$IND(P) = \{(x, y) \in U \times U | \forall a \in P, f(x, a) = f(y, a)\}. \quad (1)$$

$IND(P)$ is also called the indiscernibility relation with respect to $P$. If $(x, y) \in IND(P)$, the objects $x$ and $y$ are said to be indiscernible from each other by the attributes from $P$. It is easy to show that $IND(P)$ is an equivalence relation on $U$ and $IND(P) = \bigcap_{a \in P} IND(\{a\})$. The relation $IND(P)$ induces a partition of $U$, denoted by $U/IND(P) = \{[x]_P | x \in U\}$, where $[x]_P$ denotes the equivalence class determined by $x$ with respect to $P$, i.e., $[x]_P = \{y \in U | (x, y) \in IND(P)\}$.

As any equivalence relation induces a partition of the universe, these partitions can be used to build new subsets of the universe. These notions can be formally expressed as follows.

Let $DT = (U, A, V, f)$ be a categorical data table, $P \subseteq A$ and $X \subseteq U$. One can approximate $X$ using only the information in $P$ by constructing the lower approximation and the upper approximation of $X$, denoted as $\underline{P}X$ and $\overline{P}X$ respectively, where $\underline{P}X = \{x | [x]_P \subseteq X\}$ and $\overline{P}X = \{x | [x]_P \bigcap X \neq \emptyset\}$.

The objects in $\underline{P}X$ can be classified with certainty as members of $X$ on the basis of knowledge in $P$, while the objects in $\overline{P}X$ can only be classified as possible members of $X$. The set $BN_P(X) = \overline{P}X - \underline{P}X$ is called the $P$-boundary region of $X$, and consists of those objects that cannot be decisively classified into $X$ on the basis of knowledge in $P$. The set $U - \overline{P}X$ is called the $P$-outside region of $X$ and consists of those objects which can not belong to $X$ certainly. A set is said to be rough if the boundary region is non-empty.

A rough set can be characterized numerically by the following term

$$\alpha_P(X) = \frac{|\underline{P}X|}{|\overline{P}X|}. \tag{2}$$

which is called the approximation accuracy, where $|X|$ denotes the cardinality of $X \neq \emptyset$. Obviously, $0 \leq \alpha_P(X) \leq 1$. If $\alpha_P(X) = 1$, $X$ is said to be crisp with respect to $P$, i.e., $X$ is precise with respect to $P$. Otherwise, if $\alpha_P(X) < 1$, $X$ is said to be rough with respect to $P$, i.e., $X$ is vague with respect to $P$.

### 2.2 Problem Statement

Similarly, a categorical time-evolving data can also be stored in a table. Formally, a categorical time-evolving data table [5] can be formulated as a quintuple $TDT = (U, A, V, f, t)$ , where $U$, $A$ and $V$ are the same as those in $DT$. The information function $f : U \times A \times t \to V$ is a mapping such that for any $x \in U$ and $a \in A$, $f(x, a, t) \in V_a$, where $t$ is the arriving time of object $x$. As the arrival objects change with time, concepts often change at different time points. In order to detect the change of concepts, we adopt the sliding window technique which is used in the numerical data streams [6–8] to partition a categorical time-evolving data table. Suppose that $N$ is the sliding window size, then the $TDT$ is separated into several continuous subsets $S^{T_i} (1 \leq i \leq \lfloor \frac{U}{N} \rfloor)$ and each subset $S^{T_i}$ has $N$ objects. Each subset can also be called a concept. The superscript number $T_i$ is the identification number of the sliding window and $T_i$ is also called timestamp. In this work, our goal is to detect the difference between $S^{T_{i+1}}$ and $S^{T_i}$ and analyze the speed of the difference.

## 3 Concept-Drifting Detecting

In this section, we define the lower approximation and upper approximation of a set $Y$ with respect to a data set $X$ instead of a universe $U$ in rough set theory. To enable quantitative analysis of concept drifting for categorical evolving data, we formulate a set of measures for changes of concepts, including the degrees and speeds of a new concept emerging and a old concept emerging as well as the speed of change between two concepts.

### 3.1 Measures of Concepts Changes

To formulate the change of a concept, we define the lower approximation and upper approximation of a set as

**Definition 1.** *Let $TDT = (U, A, V, f, t)$ be a categorical time-evolving data table, $P \subseteq A$ and $X \subseteq U$. For any $Y \subseteq X$ and $x \in X$, the lower approximation and upper approximation of $Y$ with respect to $X$ are defined as*

$$\underline{P}Y = \{x | [x]_P \subseteq Y\} \tag{3}$$

*and*

$$\overline{P}Y = \{x | [x]_P \bigcap Y \neq \emptyset\}, \tag{4}$$

*where $[x]_P = \{y \in X | (x, y) \in IND(P)\}$.*

Here, the lower approximation and upper approximation of $Y$ are defined with respect to $X$, not to the universe $U$.

Given a categorical data steam that carries a set of concepts at different time points, at a particular time point, a concept contains a set of sub-concepts and the concept changes as sub-concepts change over time. For example, in social media data streams, a topic may consist of several subtopics at a given time point and the topic changes as a new subtopic emerges or an old subtopic disappears at the following time points. We use an intuitive example in Fig.1 to illustrate three types of concept change.

Assume the two rectangles in each sub figure of Fig.1 represent a concept at two consecutive time points $t_1$ and $t_2$ from left to right. Each rectangle contains two or three sub-concepts described by circles in different colors. Fig.1(a) shows the yellow sub-concept emerged at $t_2$ after $t_1$. Fig.1(b) shows yellow sub-concept disappeared at $t_2$ from the concept. In Fig.1(c), two old sub-concepts faded and two new sub-concepts emerged.

Using the definitions of lower approximation and upper approximation in Definition 1, we define the measures for degrees and speeds of new concept emerging and old concept fading in categorical evolving data as follows.

**Definition 2.** *Let $TDT = (U, A, V, f, t)$ be a categorical time-evolving data table and $S^{T_i}, S^{T_j} \subseteq U$, where $S^{T_i} \bigcap S^{T_j} = \emptyset$ and $S^{[T_i, T_j]} = S^{T_i} \bigcup S^{T_j}$. The new*

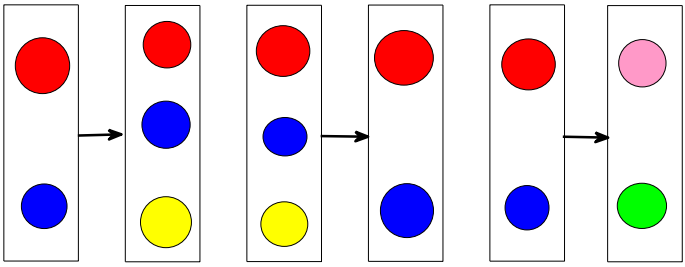


**Fig. 1.** Three types of concept change

*concepts emerging degree and old concepts fading degree from* $S^{T_i}$ *to* $S^{T_j}$ *with respect to* $A$ *are defined as*

$$NED_A < S^{T_i}, S^{T_j} > = \frac{1}{|A|} \sum_{a \in A} NED_{\{a\}} < S^{T_i}, S^{T_j} > \tag{5}$$

*and*

$$OFD_A < S^{T_i}, S^{T_j} > = \frac{1}{|A|} \sum_{a \in A} OFD_{\{a\}} < S^{T_i}, S^{T_j} >, \tag{6}$$

*where*

$$NED_{\{a\}} < S^{T_i}, S^{T_j} >= \frac{|\underline{\{a\}} S^{T_j}|}{|\overline{\{a\}} S^{T_j}|},$$

$$OFD_{\{a\}} < S^{T_i}, S^{T_j} >= \frac{|\underline{\{a\}} S^{T_i}|}{|\overline{\{a\}} S^{T_i}|}.$$

*Here,* $\underline{\{a\}} S^{T_m} (m = i, j)$ *represents the lower approximation of* $S^{T_m}$ *in* $S^{[T_i,T_j]}$ *with respect to attribute* $a$*, and* $\overline{\{a\}} S^{T_m} (m = i, j)$ *represents the upper approximation of* $S^{T_m}$ *in* $S^{[T_i,T_j]}$ *with respect to attribute* $a$*.*

$NED_A < S^{T_i}, S^{T_j} >$ and $OFD_A < S^{T_i}, S^{T_j} >$ are used to measure the degrees of concept change between two consecutive time points. The higher the value of $NED_A < S^{T_i}, S^{T_j} >$ or $OFD_A < S^{T_i}, S^{T_j} >$ is, the more dramatic the change of a concept from $S^{T_i}$ to $S^{T_j}$, either a sub-concept emerged or faded.

According to Eq.(1), the degree of a concept change with respect to an attribute $a$, $NED_{\{a\}} < S^{T_i}, S^{T_j} >$ or $OFD_{\{a\}} < S^{T_i}, S^{T_j} >$ equals to 1 if $S^{[T_i,T_j]}/IND(\{a\}) = \{X | X = \{u\}, u \in S^{[T_i,T_j]}\}$. The degree of a concept change with respect to an attribute $a$ equals to 0 if $S^{[T_i,T_j]}/IND(\{a\}) = \{X | X = S^{[T_i,T_j]}\}$. In other situations, $0 < NED_{\{a\}} < S^{T_i}, S^{T_j} >, OFD_{\{a\}} < S^{T_i}, S^{T_j} > \ < 1$. Therefore, we have $0 \leq NED_A < S^{T_i}, S^{T_j} >, OFD_A < S^{T_i}, S^{T_j} > \leq 1$.

The speed of concept drifting was used [9]. In this paper, we use speed to measure the amount of concept change from $t_1$ to $t_2$. The speeds of new concept emerging and old concept fading are defined as follows.

**Definition 3.** *Let $TDT = (U, A, V, f, t)$ be a categorical time-evolving data table and $S^{T_i}, S^{T_j} \subseteq U$, where $S^{T_i} \bigcap S^{T_j} = \emptyset$ and $S^{[T_i,T_j]} = S^{T_i} \bigcup S^{T_j}$. The new concepts emerging speed and old concepts fading speed from $S^{T_i}$ to $S^{T_j}$ with respect to $A$ are defined as*

$$NES_A < S^{T_i}, S^{T_j} > = NED_A < S^{T_i}, S^{T_j} > \times \frac{|S^{T_j}|}{t_j} \tag{7}$$

*and*

$$OFS_A < S^{T_i}, S^{T_j} > = OFD_A < S^{T_i}, S^{T_j} > \times \frac{|S^{T_i}|}{t_i}. \tag{8}$$

*where $\frac{|S^{T_m}|}{t_m} (m = i, j)$ represents the flowing speed of objects.*

By considering the degrees of new concept emerging and old concept fading together, we define the degree and speed of change between two concepts as:

**Definition 4.** *Let $TDT = (U, A, V, f, t)$ be a categorical time-evolving data table and $S^{T_i}, S^{T_j} \subseteq U$, where $S^{T_i} \bigcap S^{T_j} = \emptyset$ and $S^{[T_i,T_j]} = S^{T_i} \bigcup S^{T_j}$. The degree and speed of change between $S^{T_i}$ and $S^{T_j}$ with respect to $A$ are defined respectively as*

$$CD_A(S^{T_i}, S^{T_j}) = \frac{NED_A < S^{T_i}, S^{T_j} > + OFD_A < S^{T_i}, S^{T_j} >}{2} \tag{9}$$

*and*

$$CS_A(S^{T_i}, S^{T_j}) = CD_A(S^{T_i}, S^{T_j}) \times \frac{|S^{[T_i,T_j]}|}{t_i + t_j}. \tag{10}$$

It is easy to prove that $CD_A(S^{T_i}, S^{T_j})$ is a metric.

*Property 1.* Let $TDT = (U, A, V, f, t)$ be a categorical time-evolving data table. For any $S^{T_i}, S^{T_j}, S^{T_k} \subseteq U$, where $S^{T_i} \bigcap S^{T_j} \bigcap S^{T_k} = \emptyset$, we have

(1) Symmetry: $CD_A(S^{T_i}, S^{T_j}) = CD_A(S^{T_j}, S^{T_i})$;
(2) Nonnegativity: $CD_A(S^{T_i}, S^{T_j}) \geq 0$; and
(3) Triangle Inequality: $CD_A(S^{T_i}, S^{T_j}) + CD_A(S^{T_j}, S^{T_k}) \geq CD_A(S^{T_i}, S^{T_k})$.

*Example 1.* We use the simple categorical time-evolving data set in Table 1 to show the procedure of computing the degree of change between two concepts. The speed of change can be computed similarly.

In Table 1, data set is $X = \{x_1, x_2, \ldots, x_{20}\}$ and $A = \{A_1, A_2, A_3\}$ is the attribute set. Assume there are 5 records in each sliding window (i.e., the window size N=5), and totally 4 windows in $X$, i.e., $S^{T_1} = \{x_1, x_2, \cdots, x_5\}$, $S^{T_2} = \{x_6, x_7, \cdots, x_{10}\}$, $S^{T_3} = \{x_{11}, x_{12}, \cdots, x_{15}\}$ and $S^{T_4} = \{x_{16}, x_{17}, \cdots, x_{20}\}$.

Using Definition 1, we calculate

$$S^{[T_1,T_2]}/IND(\{A_1\}) = \{\{x_1, x_5, x_6, x_8, x_{10}\}, \{x_2, x_4, x_9\}, \{x_3, x_7\}\},$$

$$S^{[T_1,T_2]}/IND(\{A_2\}) = \{\{x_1, x_4, \cdots, x_{10}\}, \{x_2, x_3\}\},$$

$$S^{[T_1,T_2]}/IND(\{A_3\}) = \{\{x_1, x_6, x_{10}\}, \{x_2, x_3, x_4, x_7, x_9\}, \{x_5, x_8\}\}.$$

According to Definition 2, we calculate

**Table 1.** A categorical time-evolving data table

| Object | $A_1$ | $A_2$ | $A_3$ |
|---|---|---|---|
| $x_1$ | A | M | C |
| $x_2$ | Y | E | P |
| $x_3$ | X | E | P |
| $x_4$ | Y | M | P |
| $x_5$ | A | M | D |
| $x_6$ | A | M | C |
| $x_7$ | X | M | P |
| $x_8$ | A | M | D |
| $x_9$ | Y | M | P |
| $x_{10}$ | A | M | C |
| $x_{11}$ | B | E | G |
| $x_{12}$ | X | M | P |
| $x_{13}$ | B | E | D |
| $x_{14}$ | Y | M | P |
| $x_{15}$ | B | F | D |
| $x_{16}$ | Y | M | P |
| $x_{17}$ | X | M | P |
| $x_{18}$ | Z | N | T |
| $x_{19}$ | X | M | P |
| $x_{20}$ | Y | M | P |

$$NED_{\{A_1\}} < S^{T_1}, S^{T_2} >= \frac{|\emptyset|}{|\{x_1, x_2, \cdots, x_{10}\}|} = 0,$$

$$NED_{\{A_2\}} < S^{T_1}, S^{T_2} >= \frac{|\emptyset|}{|\{x_1, x_4, \cdots, x_{10}\}|} = 0,$$

$$NED_{\{A_3\}} < S^{T_1}, S^{T_2} >= \frac{|\emptyset|}{|\{x_1, x_2, \cdots, x_{10}\}|} = 0,$$

$$OFD_{\{A_1\}} < S^{T_1}, S^{T_2} >= \frac{|\emptyset|}{|\{x_1, x_2, \cdots, x_{10}\}|} = 0,$$

$$OFD_{\{A_2\}} < S^{T_1}, S^{T_2} >= \frac{|\{x_2, x_3\}|}{|\{x_1, x_2, \cdots, x_{10}\}|} = \frac{1}{5},$$

$$OFD_{\{A_3\}} < S^{T_1}, S^{T_2} >= \frac{|\emptyset|}{|\{x_1, x_2, \cdots, x_{10}\}|} = 0,$$

Using Definition 3 and Definition 4, we obtain

$$\begin{aligned} CD_A(S^{T_1}, S^{T_2}) &= \tfrac{1}{2} \times (NED_A < S^{T_1}, S^{T_2} > + OFD_A < S^{T_1}, S^{T_2} >) \\ &= \tfrac{1}{2} \times \tfrac{1}{3}(0 + 0 + 0 + \tfrac{1}{5} + 0) \\ &= 0.0333 \end{aligned}$$

With similar computations, we obtain

$$CD_A(S^{T_2}, S^{T_3}) = 0.2507$$

$$CD_A(S^{T_3}, S^{T_4}) = 0.2381$$

We can compare the degrees of change at consecutive windows as

$$CD_A(S^{T_1}, S^{T_2}) < CD_A(S^{T_3}, S^{T_4}) < CD_A(S^{T_2}, S^{T_3}).$$

If we set 0.2 as a threshold, we can identify that concept has drifted from $S^{T_2}$ to $S^{T_3}$ and from $S^{T_3}$ to $S^{T_4}$. $S^{T_3}$ and $S^{T_4}$ are considered as concept drifting windows.

If $t_1, t_2, t_3, t_4$ are the duration times of the 4 sliding windows, we can compute the speeds of changes $NES_A$, $OFS_A$ and $CS_A$ between consecutive sliding windows using Definition 3 and Definition 4, as shown in Table 2.

**Table 2.** The change speed between consecutive sliding windows

| Sliding windows | $NES_A$ | $ODS_A$ | $CS_A$ |
|---|---|---|---|
| $S^{T_1} \longrightarrow S^{T_2}$ | 0 | $0.0667 \times \frac{5}{t_1}$ | $0.0667 \times \frac{10}{t_1+t_2}$ |
| $S^{T_2} \longrightarrow S^{T_3}$ | $0.2845 \times \frac{5}{t_3}$ | $0.2169 \times \frac{5}{t_2}$ | $0.2507 \times \frac{10}{t_2+t_3}$ |
| $S^{T_3} \longrightarrow S^{T_4}$ | $0.1429 \times \frac{5}{t_4}$ | $0.3333 \times \frac{5}{t_3}$ | $0.2381 \times \frac{10}{t_3+t_4}$ |

### 3.2 Concept-Drifting Detecting Algorithm

From the above definitions, drifting of a concept can be detected by comparing the degree of change against a given threshold. As a result, a concept-drifting detection algorithm $CDDA$ is developed as shown in Algorithm 1. The key step of $CDDA$ is to compute the degree of change between two consecutive sliding windows $CD_A(S^{T_i}, S^{T_{i+1}})$. The complexity of this computation is $O(|S^{[T_i,T_{i+1}]}|^2|A|)$. Therefore, the time complexity of $CDDA$ algorithm is $O(\lfloor \frac{|X|}{N} \rfloor |S^{[T_i,T_{i+1}]}|^2|A|) = O(\lfloor \frac{|X|}{N} \rfloor 4N^2|A|) = O(|X|N|A|)$, where $X$ is the data set, $|A|$ the number of attributes, and $N$ the size of sliding windows. We can see that the time complexity of $CDDA$ is linear with respect to the number of the objects in $X$.

## 4 Experimental Results and Analysis

### 4.1 Data Set

We used the 10% subset version of the KDD-CUP'99 Network Intrusion Detection stream data set [10] to test the $CDDA$ algorithm. The Network Intrusion

**Algorithm 1.** The concept-drifting detection algorithm

**Input:**
- $TDT = (U, A, V, f, t)$ : the data set,
- $N$ : the size of sliding window,
- $\theta$ : the specified threshold value,

**Output:** Driftingwindow;
**Method:**
Driftingwindow=$\emptyset$;
**for** $i = 1$ *to* $\lfloor \frac{|U|}{N} \rfloor - 1$ **do**
  **if** $CD_A(S^{T_i}, S^{T_{i+1}}) \geq \theta$ **then**
    Driftingwindow=Driftingwindow $\bigcup\{i+1\}$;
  **end if**
**end for**

Detection data set consists of a series of TCP connection records from two weeks of LAN network traffic data managed by MIT Lincoln Labs. Each record corresponded to either a normal connection or an intrusion (or attack). The attacks include 22 types. In the following experiments, all 22 attack-types are seen as "attack". In this data set, there are 494,021 records and each record contains 42 attributes (class label is included). We discretized the 34 numerical attributes using the uniform quantization method and each attribute was quantized into 5 discrete values.

### 4.2 Concept-Drifting Detection

The size of the sliding windows and the given threshold are two parameters that affect the detection of concept drifting. We conducted a series experiments to investigate the settings of these two parameters. The experiment results are presented as follows.

**Experiment 1.** In this experiment, the threshold was set to 0.01 and the size of the sliding windows changed from 1000 to 30000 with a step length of 1000. The variations of the number of drifting-concepts with respect to the class label and the attribute set are shown in Fig.2.

From Fig.2, we can see that the number of drifting-concepts decreased with increase of the sliding window size.

**Experiment 2.** In this experiment, the size of the sliding window was set to 3000 and the threshold changed from 0.01 to 1 with a step length of 0.01. The number of drifting-concepts changed as threshold changed with respect to the class label and the attribute set. The result is shown in Fig.3.

From Fig.3, we can see that the change rate on the number of drifting-concepts over the threshold with respect to the attribute set is greater than that with respect to the class label. To make the number of drifting-concepts with respect to the class label as close as possible to the number with respect to the attribute set, the threshold with respect to the class label should be greater than the

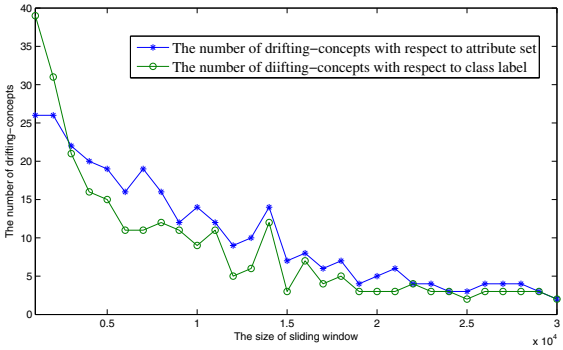


**Fig. 2.** The number of drifting-concepts varying with the size of the sliding windows

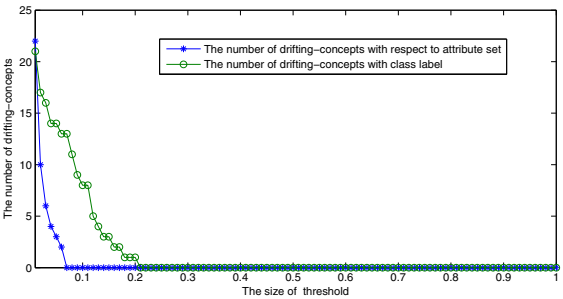


**Fig. 3.** The number of drifting-concepts varying with the values of the threshold

threshold with respect to the attribute set. In practice, a user can choose a threshold according to a prior knowledge or specific requirement.

**Experiment 3.** The duration of objects was assumed same in each sliding window and the evolving speeds of concepts in different sliding windows are shown in Fig.4. In this experiment, the size of the sliding window was set to 3000.

In Fig.4, the values of the change speed drop to zero in the range of 51 to 114, 134 to 149, and 155 to 160 because the records are same in these sliding windows of each interval.

## 5 Related Work

Detection of concept drifting has become an interesting research topic recently. The problem of detecting concept drifts in numerical data was explored in [11, 12]. As for detection of concept drifting in categorical data, a method was proposed to determine concept drifts by measuring the difference of cluster distributions between two continuous sliding windows from categorical data streams

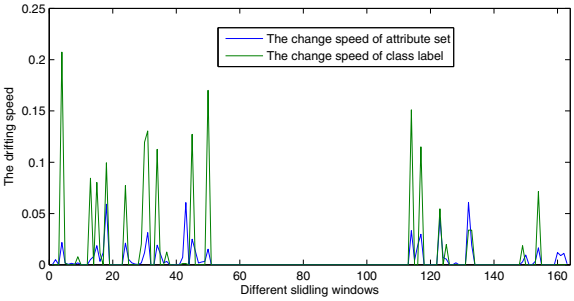


**Fig. 4.** The evolving speed on KDD-CUP'99 data set

[13]. The shortcoming of the method is the difficulty to set suitable system parameters for different applications. In [14], a framework was presented for detecting the change of the primary clustering structure which was indicated by the best number of clusters in categorical data streams. However, setting the decaying rates to adapt to different types of clustering structures is very difficult. Nasraoui [15] presented a framework for mining, tracking, and validating evolving multifaceted user profiles which summarize a group of users with similar access activities. In fact, two continuous sliding windows can be considered as two concepts. Cao [5] used rough set theory to define the distance between two concepts as the difference value of the degree of membership of each object belonging to two different concepts, respectively. This method only requires one parameter to set, so it is easy to use in real applications. However, the distance can only detect the change of concepts, and reasons that cause the change are not considered.

## 6 Conclusion

In this paper, based on sliding window techniques and approximation accuracy, the change degree and the change speed of concepts have been defined, and a concept-drifting detection algorithm has been proposed. The time complexity analysis and experimental results on a real data set have demonstrated the proposed algorithm is not only effective in detecting concept drifts from categorical data streams but also efficient in processing large data sets due to its linearity with respect to input data $X$.

**Acknowledgements.** This work is supported by Shenzhen Internet Industry Development Fund under Grant JC201005270342A, China Postdoctoral Science Foundation under Grant 2012M510046, the Natural Science Foundation of Shanxi under Grant 2010021016-2.

## References


1. Babcock, B., Babu, S., Dater, M., Motwanti, R.: Models and Issues in data stream systems. In: Proc. PODS, pp. 1–16 (2002)
2. Widmer, G., Kubat, M.: Learning in the presence of concept drift and hidden context. Machine Learning 23, 69–101 (1996)
3. Guha, S., Meyerson, A., Mishra, N., Motwani, R., OCallaghan, L.: Clustering data streams: theory and practice. IEEE Transactions Knowledge and Data Engineering 15, 515–528 (2003)
4. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences 11, 341–356 (1982)
5. Cao, F.Y., Liang, J.Y., Bai, L., Zhao, X.W., Dang, C.Y.: A framework for clustering categorical time-evolving data. IEEE Transactions on Fuzzy Systems 18, 872–885 (2010)
6. Aggarwal, C.C., Han, J., Wang, J., Yu, P.S.: A framework for clustering evolving data streams. In: Proc. Very Large Data Bases Conf. (2003)
7. Chakrabarti, D., Kumar, R., Tomkins, A.: Evloluationary clustering. In: Proc. ACM SIGKDD. Knowledge Discovery and Data Mining, pp. 554–560 (2006)
8. Gaber, M.M., Yu, P.S.: Detection and classification of changes in evolving data streams. International Journal of Information Technology and Decision Making 5, 659–670 (2006)
9. Minku, L.L., White, A.P., Yao, X.: The impact of diversity on online ensemble learning in the presence of concept drift. IEEE Transactions on Knowledge and Data Engineering 22, 730–742 (2010)
10. UCI Machine Learning Repository (2012), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
11. Dai, B.-R., Huang, J.-W., Yeh, M.-Y., Chen, M.-S.: Adaptive clustering for multiple evolving steams. IEEE Transactions Knowledge and Data Engineering 18, 1166–1180 (2006)
12. Yeh, M.Y., Dai, B.R., Chen, M.S.: Clustering over multiple evolving streams by events and corrlations. IEEE Transactions Knowledge and Data Engineering 19, 1349–1362 (2007)
13. Chen, H.-L., Chen, M.-S., Lin, S.-C.: Catching the trend: A framework for clustering concept-drifting categorical data. IEEE Transactions Knowledge and Data Engineering 21, 652–665 (2009)
14. Chen, K.K., Liu, L.: HE-Tree:a framework for detecting changes in clustering structure for categorical data streams. The VLDB Journal 18, 1241–1260 (2009)
15. Nasraoui, O., Soliman, M., Saka, E., Badia, A., Germain, R.: A web usage mining framework for mining evolving user profiles in dynamic web sites. IEEE Transactions Knowledge and Data Engineering 20, 202–215 (2008)