PROCEEDINGS

CYBERNETICSCOM 2017

The 2017 IEEE International Conference on Cybernetics and Computational Intelligence

Phuket, Thailand

20-22 November 2017

IEEE The Indonesia Section Computer Society Chapter

IEEE INDONESIA SECTION

BINUS UNIVERSITY DOCTORATE PROGRAM

Prince of Songkla University

Table of Content

Survey of Emerging Patterns

Nizirwan Anwar
Faculty of Computer Science
Esa Unggul university
Jakarta, Indonesia
nizirwan.anwar@esaunggul.ac.id

Harco Leslie Hendric Spits Warnars
Computer Science Department, BINUS Graduate Program – Doctor of Computer Science
Bina Nusantara University
Jakarta, Indonesia 11480
Spits.hendric@binus.ac.id

Sasmoko
Primary Teacher Education Department, Faculty of Humanities, Bina Nusantara University, Jakarta, Indonesia 11480
sasmoko@binus.edu

Horacio Emilio Perez Sanchez
Structural Bioinformatics and High Performance Computing Research Group (BIO-HPC)
Universidad Católica de Murcia (UCAM)
Guadalupe, Spain
hperez@ucam.edu

***Abstract*—Emerging patterns (EPs) which found in 1999 has been proven as strong discriminator which strongly describe significant between 2 datasets. As strong discriminator, EPs will be interested to be used, applied and mixed in many algorithms for finding patterns in many different datasets particularly for text datasets. Using EPs algorithm in Database Management Systems (DBMS) such as MySQL, SQLServer and etc will be interested as well and need to be explored. The differences between 2 datasets literally discriminate knowledge between those datasets which represent with growthrate number as justification of EPs. Moreover, confidence of EPs can be measured in order to secure of finding EPs where confidence will have 100% as maximum score. Since the discrimination is not only between 2 datasets then EPs algorithms have been extended to discriminate between more than 2 datasets which recognized as EPs classification and there are many EPs classification algorithms including Jumping EPs classification as well.**

Keywords—Data Mining; Jumping Emerging Patterns; Discriminating Emerging Patterns; Classification Emerging Patterns;

I. Introduction

Emerging Patterns (EPs) is discovery knowledge from database capture emerging trending when applied in timestamped databases or capture useful contrasts between data classes when applied to datasets. Moreover, EPs capture significant changes or differences between two or more than two datasets are defined as itemsets whose supports (frequencies) increase significantly from one to another dataset. The changing of supports for itemsets from one to another dataset as ratios of two supports will be called growth rates. Furthermore, EPs use user-defined threshold in order to reduce large candidate patterns, then can be said EPs are itemsets or patterns which growth rates are larger than a specific assigned threshold. Finally, EPs are similar to discriminant rules or evolution rules in Attribute Oriented Induction (AOI) [37,38,39] but different since EPs do not confined by exclusiveness of constraint and because the extra information of growth rate [1].

The EPs with very high growth rates are notable differentiating characteristic between 2 datasets and useful for building powerful classification [1,27]. Thus, Those EPs with very large growth rates are frequent in one of class but rare in another. Meanwhile EPs between low and medium supports such as between 1% and 20% can give very useful new insights and become guidance for experts, in even “well understood” applications [1]. Hence, the low supports EPs such as 0.1 until 5% may be new knowledge to the dataset and discover small support EPs is interesting [1]. The interestingness of discovery small support EPs due to reason too many EPs candidates and make naive algorithms too costly to examine all itemsets in dataset. For example if there are 350 itemsets in dataset then naive algorithm would need to process 2^{350} (Cartesian product) itemsets in order to find their supports in datasets D1 and D2 and then determine their finding growth rates. Overall, EPs algorithm can be divided become process discriminating between two datasets and classification more than two datasets.

II. Current EPs Implementation

A number of researches are carried out on EPs and EPs was applied to raster geospatial dataset[28], and the difference of between safe and non-safe power load line is applied by Piao et al with EPs and proposes an incremental Temporal Frequent Patterns, TFP-tree algorithm for mining EPs that can perform efficiently within memory limitation [29]. Maintain the high interpretability of EPs and offer a high prediction performance is the advantages of EP with PolyA-iEP algorithm [30]. EPs was chosen due to efficiency in finding of EPs from tens of thousand transactions in seconds in visual data by extending EPs to discriminate features including their combination in-between different actions in video[31]. Classification algorithm called Frequent Emerging Sequence Patterns (FESP) was applied to define new support and growth rate of support from DNA sequence database to find frequent EPs [32]. Meanwhile, The equation of EPs was embedded to AOI-HEP in order to discriminate 2 datasets in order to find frequent and similar AOI-HEP patterns [41,42,43].

EPs has been used and combined with other algorithms like EPs are combined with Decision tree technique CART-based algorithm in order to find EPs for classification that consist six steps where CART trees replace border-based algorithm function [45]. A user-friendly tool KTDA system is CART-based method implementation with some extensions and improvements [44]. Moreover, EPs are used to construct weighted Support Vector Machines (weightedSVMs) by calculating numeric scores for each instance based on EPs then use scores to assign weights for training [46]. Next, Fuzzy

SVM classfifier algorithm uses EPs algorithm to weight the training instances for the class membership[47]. Generalize decision tree and weighted classes are assigned to train data instances and discover weights for training instances in decision tree[48]. Meanwhile, proposed EP-weighting scheme algorithm as visual word weighting scheme by finding EPs for visual keywords in the training dataset and for each visual word according to EPs by performance of adaptive weighting assignment[49]. Moreover, proposed Contrast Pattern tree (CP-tree) algorithm was inspired by FP-Tree which mining frequent patterns without candidate generation, for mining Strong Jumping EPs (SJEPs), Noise-tolerant EPs (NEPs) and Generalized Noise-tolerant EPs (GNEPs) for classification task [53]. Furthermore, EPs and Decision Tree algorithms are used in rare-class classification (EPDT) where EPs is used to improve the quality of rare-case classification. [27,50]. Next, EPs algorithm are used in rare-class classification (EPRC) algorithm with three stages such as: firstly, rare class from undiscovered EPs is generated, and next support of rare-class EPs is increasing including pruning low support EPs [27,51]. Lastly, EPs algorithm and Generic Method (GM) are combined to expand the space in training data(ETDS) with four methods such as generation by superimposing EPs, generation by Crossover, generation by Mutation and generation by Mutation and EPs [52].

Meanwhile, EPs had been implemented in table of Database Management Systems (DBMS) particularly of relational table databases. MRDM (Multi Relational Data Mining) method [57,58,59] or can be called Mr-EP (Multi Relational Emerging Pattern) [54,55,56] are EPs algorithm which discovers EPs from data scattered in multiple tables of a relational database. Previous algorithms before this algorithms assume that data to be mined are stored in a single data table and since previous dataset was saved in text file dataset then process upon table DBMS will have different treatment processes [54,55,56].

III. Growth Rate, Support and Confidence

EPs are defined with GrowthRate score as shown in equation (1) where EPs are associated with two datasets, like dataset D1 will be called background dataset or can be called negative class of the EPs and dataset D2 will be called target dataset or can be called positive class.

$$\text{GrowthRate(X)} = \frac{target}{background} = \frac{Positive}{Negative} = \frac{supp_{D2}(X)}{supp_{D1}(X)} \quad (1)$$

Assume there are given an ordered pair of datasets D1 and D2 then growth rate of an itemset X from datasets D1 to D2 denoted in equation 1 or 2 as GrowthRate D1→D2(X) = $supp_{D2}(X)$/ $supp_{D1}(X)$.

$$\text{GrowthRate(X)} = \frac{supp_{D2}(X)}{supp_{D1}(X)} = \frac{\frac{count_{D2}(X)}{|D2|}}{\frac{count_{D1}(X)}{|D1|}} \quad (2)$$

where :

∞ = infinity, when (n/0), Jumping EPs (JEPs)

suppD1(X) = support in dataset D1 containing itemset X

suppD2(X) = support in dataset D2 containing itemset X

GrowthRate in equation (2) can have 3 different results and they are:

1. GrowthRate=0, if suppD2(X)=0
2. GrowthRate=∞ if suppD1(X)=0 and suppD2(X) ≠0.
3. GrowthRate>0,if given ρ> 1 as growth rate threshold, an itemset X is said to be an ρ-emerging pattern (ρ-EPs or simply EPs) from D1 to D2 (sometimes states as an EPs in/of D2) if GrowthRate(X)≥ρ.

EPs have different way to mention it and as shown in equation 2 where EPs of pattern/itemset X, EPs(X) = GrowthRate(X)= $supp_{D2}(X)/supp_{D1}(X)$ and the way to mention EPs are [1,13,40]:

1. EPs(X) are GrowthRate(X) from $supp_{D1}(X)$ to $supp_{D2}(X)$.
2. EPs(X) are GrowthRate(X) from $count_{D1}(X)/|D1|$ to $count_{D2}(X)/|D2|$.
3. EPs(X) are GrowthRate(X) of $supp_{D2}(X)$.
4. EPs(X) are GrowthRate(X) of $count_{D2}(X)/|D2|$.
5. EPs(X) are GrowthRate(X) from Dataset D1 to Dataset D2.
6. EPs(X) are GrowthRate(X) of Dataset D2(X).
7. EPs(X) are GrowthRate(X) of the ratio support X in Dataset D2 to support X in Dataset D1.
8. EPs(X) are number of itemset (X) in $supp_{D2}(X)$ is GrowthRate times the number of itemset (X) in $supp_{D1}(X)$.
9. EPs(X) are number of itemset (X) in Dataset D2(X) is GrowthRate times the number of itemset (X) in Dataset D1(X)

A dataset is a set D of transactions. An itemset X is a subset of I, where I = {i1,i2,...,iN} be a set of items. The support of an itemset X in a dataset D, denoted as suppD(X) in equation 3.

$$\text{suppD(X)} = \frac{count_D(X)}{|D|} \quad (3)$$

where :

D = Dataset

X = Itemset or pattern, X ⊆ t, where t are instances in D

$|D|$ = total number of instances in dataset D

suppD(X) = support in dataset D containing itemset X

countD(X) = the number of transactions in dataset D containing itemset X, where countD(X) = t ∈ D and X ⊆t, where t are instances in D.

Growth rate of an EPs as shown in equation 1 and 2 can have the confidence of predictions such as equation (4), (5) or (6) that one can justify the confidence of EPs [17].

$$\text{Conf(X)} = \frac{GrowthRate\,(X)}{GrowthRate\,(X)+1} \quad (4)$$

$$\text{Conf(X)} = \frac{supp_{D2}(X)}{supp_{D2}(X)+supp_{D1}(X)} \quad (5)$$

$$\text{Conf(X)} = \frac{GrowthRate\,(X)*supp_{D1}(X)}{GrowthRate\,(X)*supp_{D1}(X)+supp_{D1}(X)} \quad (6)$$

IV. Discriminating Emerging Patterns

Figure 1 shows decomposition of the EPs mining problem which is for a given growth rate threshold ρ to find all ρ-EPs. EP mining problem can be described with the supports of all ρ-

EPs from D1 to D2 must fall onto region ACE triangle in figure 1. EPs mining problem is divided into three sub-problems and they are [1]:

1) Finding EPs in the BCDG rectangle.

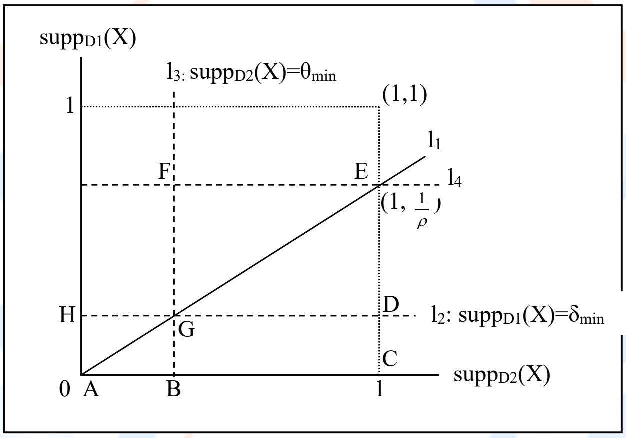
EPs in the BCDG rectangle in figure 1 are those itemsets whose supports in dataset D2 are $\geq \theta$min but in dataset D1 are $< \delta$min (suppD2(X) $\geq \theta$min and suppD1(X) $< \delta$min). The semi-naive algorithm called border-based MBD-LLBORDER algorihtm find the supports in datasets D1 and D2, all itemsets in LARGEθmin(dataset D2) and check if their growth rates are greater than ρ threshold. This semi-naive algorithm uses border LARGEδmin in dataset D1 and LARGEθmin in dataset D2 as inputs.

2) Finding EPs in GDE triangle.

EPs in the GDE triangle in figure 1 are those itemsets whose supports in datasets D2 are $\geq \theta$min and D1 are $\geq \delta$min (suppD2(X) $\geq \theta$min and suppD1(X) $\geq \delta$min). This set is exactly LARGEδmin (dataset D1) $\cap$ LARGEθmin (dataset D2).When the intersection is small we can find the EPs by checking the supports of all candidates in the intersection and when the intersection is large apply recursively the border-based MBD-LLBORDER algorithm, used for the BCDG rectangle within ACE triangle to GDE triangle until all EPs are founded.

3) Finding EPs in ABG triangle.

EPs in the ABG triangle in figure 1 are those itemsets whose supports in datasets D2 are $< \theta$min and D1 are $< \delta$min (suppD2(X) $< \theta$min and suppD1(X) $< \delta$min). EPs in ABG triangle have very small supports in datasets D1and D2 or both.



1. The Emerging Patterns support plane[1]

A. Border based algorithm

The EPs mining with border-based MBD-LLBORDER algorithm avoid the long process naive algorithms with manipulation only borders of some two datasets and derive all EPs which support satisfy a minimum support threshold in dataset D2 in order to get counting of all patterns in large of datasets[1]. The border-based MBD-LLBORDER algorithm discovers all EPs in the BCDG rectangle by calling differential procedure BORDER-DIFF algorithm in multiple numbers of times. Each call will use one itemset in the right-hand bound of the large border of dataset D2 and the whole right-hand bound of the large border of dataset D1 to form the two arguments.

BORDER-DIFF algorithm aims to derive the differential between a pair of borders with a special form: Given a pair of borders <{Ø},{U}> and <{ Ø},R1>, BORDER-DIFF algorithm derives another border <L2,{U}> such that [L2,{U}]=[{Ø},{U}]-[{Ø},R1]. There are 2 versions of BORDER-DIFF algorithms for deriving L where the first version is more declarative and thus easier to understand, and the second version is more procedural and more efficient. The improved BORDER-DIFF algorithm is more efficient because it iteratively removes non minimal elements in the intermediate result for U-S1,... ,U-Si before processing U-Si+1, thus avoiding generating large intermediate results in general and the whole cartesian products of U-S1,...,U-Sk in particular.

Meanwhile, there is previous EPs algorithm such ConsEPMiner where its a constraint based EPs Miner that utilize two types of constraints such as support and growth rate tresholds which efficiently mining EPs and use another constraint called growth rate improvement to eliminate the uninteresting EPs [33]. Beside three external constraints (support, growth rate and growth rate improvement), there are another three inherent constraints which are not user given, namely same subset support, top growth rate and same origin.

B. Previous technique before Border Based algorithm

Previous data mining techniques (association rule miner) such as max-miner[2], Apriori variant i.e. Apriori, Apriori-gen, AprioriTid (uses Apriori-gen function), AprioriHybrid(combination Apriori and AprioriTid), Apriori-like,Apriori-inspired [3,5], Apriori DHP (Direct Hashing and Pruning)[6], DIC (Dynamic Itemset Counting) [7], and simple breadth-first (levelwise)[4] only used one-sided borders on subset-closed collections. Max-miner is more efficient for dataset with long patterns [2] compare with variant Apriori algorithm [3,5]. Apriori algorithm involves a phase for finding patterns called frequent itemsets that set of items meeting a user specified threshold, but max-miner algorithm extracting only the maximal frequent itemsets. Thus, max-miner algorithm is efficiently discovery longest maximal frequent itemsets. Max-miner algorithm as extracting only the maximal frequent itemsets is similar with some algorithms like randomized [7], association rule miner algorithms like MaxEclat, MaxClique[8], pincer-search[10] (Pincer-search is combination bottom-up search in Apriori[3] and Off-line Candidate Determination (OCD) [11] and top-down search as novel). Since max-miner [2] and SE-Trees (Set Enumeration) [12] cannot be used to discover small borders then Min-miner and decreasing SE-tress are used.

C. Jumping Emerging Patterns discrimination

Jumping EPs (JEPs) are special EPs or special type of discriminant rule whose supports increase abruptly from zero support in one dataset to non-zero support in another. JEPs is EPs with infinite (∞) growth rate value whose support is zero in dataset D1 (suppD1(X)=0) and support is non-zero in dataset D2 (suppD2(X)$\neq$0). For discovering JEPs, HORIZON-MINER algorithm is used to find the large border (horizontal border) of all itemsets with non-zero support and MBD-LLBORDER is used to find JEPs using the two large borders derived by HORIZON-MINER as inputs [1]. Tree-based algorithms for computing JEPs that are 2-10 times faster than previous methods, which combination two novel features such as[21] :

1. Tree-based data structure for storing the raw data which is similar to Frequent Pattern (FP-tree) [22,23].
2. Developing of a data mining algorithm which finding patterns in the tress.

V. CLASSIFICATION EMERGING PATTERNS

Beside discriminating, then The EPs algorithms has been extended to classification called EP-based classifier where the process of finding a set of models can describe and distinguish between two or more data classes or concepts. For handling classification where distinguish more than 2 classes then each instance in dataset D is associated with p class labels: C1, C2, ..., Cp and partition dataset D into p sets: D1, D2, ..., Dp with Di containing all instances of class Ci. In this classification EPs we will discuss CAEP, iCAEP, DeEPs, BCEP and CEP. Classification by Aggregating Emerging Patterns (CAEP) is the first application of EPs to classification for datasets with more than two classes and more accurate than C4.5 and Classification Based on Association (CBA).

Algorithm classification EPs such as Classification by Aggregating EPs CAEP uses a set of EPs with multi attributes tests for each of class. CAEP algorithm for building classification EPs has 3 steps [13] and they are:

1. For each class C, where all the EPs have condition with some support and growth rate thresholds, from the opponent set of all none-C instances to the set of all C instances.
2. Aggregating the power of the discovered EPs for classifying an instance s. Aggregating differentiating score for each class C by summing the differentiating power of all EPs of class C that occur in instance s.
3. Normalizing score for class C by dividing it by some base score of the training instances of class C.

The accuracy and performance CAEP can be improved with Score Behaviour Knowledge Space (SBKS) which to record the behaviour of training data on scores to make final classification decision. SBKS is an m-dimensional space where each dimension corresponds to the score of the class [35]. Contribution of EPs and aggregate score (or score) of instances s for class C can be calculated with equation 7, where the contribution is proportional to both growth_rate(e) and suppc(e) in the target class.

$$\text{Score}(s,C) = \sum_{e\subseteq s,\, e\in E(C)} \frac{growth\_rate(e)}{growth\_rate(e)+1} * supp_c(e) \quad (7)$$

Where : s = instance
C = class
e = EPs of class C
E(C) = set of instances
growth_rate(e) = equation (2)
$supp_c(e)$ = equation (3)

There are two methods which can be used to calculate the aggregate score contributed by all the EPs of class C_i:

1. The large-border based approach, where firstly use Max-Miner algorithm [2] to discover the border of the large itemsets from D_i. If the large itemsets represented by the border can be enumerated in memory, then supports and growth rates of the EPs of C_i can be got.
2. The border differential based approach, where firstly use Max-Miner algorithm [2] to discover the two large borders of the large itemsets in D_i and the opponent D_i having certain support thresholds. Then MBD-LL_{BORDER} algorithm [1] is used to find all the EPs borders. Finally enumerate the EPs contained in EPs borders to check their supports and growth rates.

Meanwhile, EPs algorithm such as Information-based approach for classification by aggregating EPs (iCAEP) is a variant of CAEP. Compare to CAEP, iCAEP has better predictive accuracy and shorter time for training and classification [34].

Moreover, Eps algorithm like Decision making by EPs (DeEPs) [16,24,36] is a instance-based classifier which makes decisions through EPs. Instance-based approach creates remarkable reduction on both volume (the number of instances) and dimension (the number of attributes) of the training data. DeEPs have advantages on accuracy, speed and dimensional scalability over CAEP [13] and JEP-Classifier [14]. DeEPs need three main steps to determine the class of a test instance and they are:

1. Discovering border representation of EPs.
   The step aims to learn discriminating knowledge from training data, reducing the data and discovering all JEPs. Assume we have classification set Dp={P1,...,Pm} of positive instances and set Dn={N1,...,Nn} of negative instances. For each T instance, the DeEPs use three procedures to discover border representation of the EPs:
   a) Intersection the training data with T: $T\cap P_1,\ldots,T\cap P_m$ and $T\cap N_1,\ldots,T\cap N_n$ in order to reduce dimension using neighbourhood-based intersection method.
   b) Selecting the maximal itemsets from $T\cap P_1,\ldots,T\cap P_m$ and similarly from $T\cap N_1,\ldots,T\cap N_n$ in order to reduce volume
   c) There are two sub procedures :
      i. Discovery of JEPs, by mining subset of T instances which occur in D_p but not in D_n. All the JEPs in D_p by taking border difference operation $[\{\emptyset\},R_p]$ - $[\{\emptyset\},R_n]$ and the other hand mining subset of T instances which occur in D_n but not in D_p. All the JEPs in D_n by taking border difference operation $[\{\emptyset\},R_n]$ - $[\{\emptyset\},R_p]$
      ii. Discovery of common EPs, by mining subset of T instances which occur in both D_p and D_n , namely commonT=$[\{\emptyset\},R_p] \cap [\{\emptyset\},R_n]$.
2. Selecting the more discriminating EPs.
   Since the number of JEPs is usually large then the most general JEPs among all JEPs will be reduced. By the most

general JEPs is mean that the proper subsets are not JEPs anymore.

3. Determining collective scores based on the selected EPs for classification.
   Determine the collective score of T instance for any specific class C by aggregating the supports of the selected EPs in class C using compact summation method.

Futhermore, EPs algorithm like CEPClassification by EPs (BCEP) is EP for classification as hybrid of the EP-based Classifier and Naive Baiyes (NB) classifier [18,25]. There are 2 kinds of interesting EPs when mining with BCEP and they are :

1. Essential EPs (eEP), are EPs with very large growth rate (typically more than 1000), enough(large) supports in the target class (usually threshold 1%) and that are contained in the left bound of the border representing EPs collection. Large growth rate show sharp discriminating power, large supports show enough coverage on the training dataset which EPs are more resistant to noise
2. Essential JEPs (EJEP) [26], are subset of JEPs which removing JEPs that contain noise and redundant information.

BCEP utilize tree-based algorithm [21] to efficiently mine the complete eEP and EJEP for each class.

Last but not least, Eps algorithm like Constrained emerging patterns (CEP) are minimal sets of items which occur $\geq \alpha$ times in one class and $\leq \beta$ times in the other [19,20], where α and β are thresholds. CEP the same with border based MBD-LLBORDER algorithm [14] to find itemsets which support $\geq \alpha$ threshold in target (D2) dataset and support $\leq \beta$ threshold in background (D1) dataset [19,20]. CEP mining can be accomplished by an extension of JEP mining in two steps and they are:

1. Represent border based algorithm where one border represent target (D2) dataset with support $\geq \alpha$ threshold and the other border represent background (D1) dataset with support $\geq \beta$ threshold. Method for mining JEPs can be applied, once the borders are computed to gain the desired patterns in the next step [19].
2. Mining the CEP by operating on the relevant borders. When β=0, CEP become JEP and when $\beta > 0$ will have greater robustness CEP.

In figure 1, CEP is in the BCDG rectangle. For classification where more than two classes (D_1, D_2, ..., D_n,) then the EPs can be found with pair-wise feature and for example EPs for class D_1 are found by comparing D_1 against the (negative) dataset $D_2 \cup D_3 \cup ... D_n$. The EPs for class D_3 are found by comparing D_3 with respect to the (negative) dataset $D_1 \cup D_2 \cup D_4 \cup .... \cup D_n$ etc. Pair-wise classification strategy is used to mine CEP with more than two datasets, where each of dataset will be treated as target class and will be compared with unioning other datasets. For example CEP for dataset D1 are found by comparing D1 against the background dataset $D2 \cup D3 \cup ... Dn$. The CEP for dataset D3 are found by comparing D3 with respect to the background dataset $D1 \cup D2 \cup D4 \cup .... \cup Dn$ etc.

A. Reduction the number of Emerging Patterns

Using two thresholds where one for support and another for growth rate. The more lower the support threshold, the higher predictive accuracy and the higher the growth rate threshold, the higher predictive accuracy the classification achieves [13]. The chosen support threshold between 1 and 3, and growth rate threshold is 5 [13] make stable in predictive accuracy for classification.

Finding EPs in class C will have preference for EPs that have relatively large supports and growth rates, but correspond to larger coverage and stronger differentiating power. The larger coverage EPs can be removed without loss of too much accuracy and reduction the number of EPs can increase understandability of classification and predictive accuracy. The EPs reduction is optional and should not be done if it leads to poor classification of the training instances. The EPs reduction uses factors such as:

1. The absolute strength of EPs.
   Measurement using new growth rate threshold which should be larger than the EPs growth rate threshold ρ, and the measurement by selecting the strong EPs and remove the weaker EPs.
2. The relationships between EPs.
3. The relative difference between their supports and growth rates.

B. Jumping Emerging Patterns classification

JEP-Classifier is JEPs classification which partially influenced by CAEP and uses exclusively JEPs. JEP-Classifier uses datasets with more than 2 classes in an ordered way with pair-wise feature concept [14]. JEP-Classifier is aggregation of the supports of JEPs for superior classification accuracy. JEP-Classifier utilize border based algorithm to discover border of all JEPs in order to identify the most expressive JEPs. The most expressive JEPS is the most frequency JEPs which build accurate classification.

JEP-Classifier consists of four steps:

1. Discovering JEPs in dataset with semi naive or $_{\text{JEP}}$PRODUCER algorithm.
   a) Semi naive algorithm which makes limited use of borders. Which Consist two steps:
      i. Use HORIZON-MINER to discover the horizontal border of dataset D2. HORIZON-MINER algorithm is used to find the large border (horizontal border) of all itemsets with non-zero support in the dataset.
      ii. Scan dataset D1 to check the JEPs for those itemsets with zero support. SE-tree [1] algorithm can be used irredundantly and enumerate the itemsets represented by the horizontal border.

      The semi naive algorithm is slow on large databases, but not for $_{\text{JEP}}$PRODUCER algorithm.
   b) $_{\text{JEP}}$PRODUCER algorithm uses an efficient border-based algorithm.
      The inputs are two horizontal borders from datasets D1 denoted $<\{\emptyset\},R_1>$ and D2 denoted $<\{\emptyset\},R_2>$,

manipulate elements in R_1 and R_2 and produce border <L,R> to represent the set difference [{Ø},R_2] - [{Ø},R_1], namely all the JEPs in dataset D2.

2. Partition instances in dataset with pair-wise features into classes.
   The pair-wise features in dataset D, whose instances are partitioned into q classes D1,...,Dq, consist of the following q groups of JEPs: those of D1 over $\cup_{j=2}^{q}$ Dj, those of D2 over $\cup_{j\neq 2}^{q}$ Dj ,..., and those of Dq over $\cup_{j=1}^{q-1}$ Dj. For example if q=3 then pair-wise features in dataset D consist of three groups of JEPs: those of D1 over D2 υ D3, those of D2 over D1 υ D3, and those of D3 over D1 υ D2.
3. Selecting the most expressive JEPs.
   The most expressive JEPs are those JEPs with large support and can reduce its complexity and strengthen its resistance to noise in the training data. The most expressive JEPs is the left bounds of the border.
4. Determine the class labels of the test data.
   Calculate the collective impacts when a test data is given.

Another JEPs classification is JEP space where its satisfies the property of convexity and can be represented by two bounds, left bound and right bound, consisting respectively of the most general JEPs and the most specific JEPs [15,24]. For maintaining JEP space response to insertion new instances and attributes, deletion of instances and attributes. JEP space uses only one border where different with JEP-Classifier [14] which uses multiple borders. There are 3 border operations for algorithm maintaining JEP spaces and they are :

1. Border difference (-).
   Border difference is similar with MBD-LLborder algorithm [1] and using Border-Diff algorithm [1] with a slight different in output. The same like inputs for JEPPRODUCER algorithm [14], JEP space is represented with two horizontal borders (horizontal spaces or convex space) from datasets D1 of positive instances denoted <{Ø},R_1> and D2 of negative instances denoted <{Ø},R_2>. In other words, JEP space is represented with border <L,R> which have 2 bounds, they are Left bound/the most general JEPs/positive instances/<{Ø},R_1> and Right bound/the most specific JEPs/negative instances/<{Ø},R_2>. Horizontal border or horizontal space is non-zero support itemsets in the dataset. JEP space to D1 and D2 is present the set difference [{Ø},R_1] - [{Ø},R_2], where is subtracting all non-zero support itemsets in dataset D2 from all non-zero support itemsets in dataset D1.
2. Border union (υ).
   Border union is union of old JEP space and some JEP space created by new data. Suppose old JEP spaces D_1 and D_2 are positive and negative instances respectively. Assume a set i1 ($_iR_1$) of new positive instances are inserted then JEP space (D_1+i1) and D_2 or new JEP space is the union of the previous JEP space and a JEP space associated with i1. Insertion of new Left bound/the most general JEPs/positive instances/<{Ø},R_1> has set:
   ([{Ø},R_1] υ [{Ø},$_iR_1$]) - [{Ø},R_2] = ([{Ø},R_1] - [{Ø},R_2]) υ ([{Ø},$_iR_1$]) - [{Ø},R_2])
3. Border intersection (∩).
   Border intersection is intersection of old JEP space and some JEP space created by new data. Suppose old JEP spaces D_1 and D_2 are positive and negative instances respectively. Assume a set i2 ($_iR_2$) of new negative instances are inserted then JEP space D_1 and (D_2 +i2) or new JEP space is the intersection of the previous JEP space and a JEP space associated with i2. Insertion of new Right bound/the most specific JEPs/negative instances/<{Ø},R_2> has set:
   [{Ø}, R_1] – ([{Ø},R_2] υ [{Ø},$_iR_2$])=([{Ø},R_1] - [{Ø},R_2]) ∩ ([{Ø}, R_1]) - [{Ø},$_iR_2$])

Moreover, there are other JEP classifiers such as Essential JEP(EJEP) and EJEP-Classifier (EJEP-C) where for discrimination and classification respectively with adopting pair-wise feature concept[18,26]. Essential JEP (EJEP) and EJEP-Classifier (EJEP-C). EJEP is discrimination between two classes and EJEP-Classifier (EJEPC-C) is classification for more than two classes by aggregating EJEPs with adopting pair-wise features concept. EJEP-C uses two parameters: the minimum support threshold and the percentage of top ranking items used for mining EJEPs. EJEP uses tree structure called Pattern-tree (P-tree) algorithm to mine EJEPs and the method advantage is a single-scan algorithm which efficiently mine EJEPs of both data classes (from D1 to D2 and from D2 to D1) at the same time [18,26]. Whilst border-based and ConsEPMiner algorithms will call the algorithm twice using target classes D2 and D1 separately.

VI. Conclusion

Current Emerging Patterns (EPs) algorithms are only applied in text dataset and literally will be interested to be explored in Database Management Systems (DBMS) such as MySQL, Oracle, SQLServer and etc and obviously will change the way to implement EPs algorithm. Current EPs algorithms implementation can be explored as supervised pattern where user need to input and customise their data for doing discrimination and classification with EPs. Next implementation will be interested if implemented with unsupervised pattern, where the EPs algorithm will be run without user involvement, where unsupervised EPs algorithm will automatically generate all the discrimination result. As strong discriminate, EPs should suitable for any intelligent application which learning dataset based on finding pattern with activities such as discrimination or classification.

References

[1] Dong, G. and Li, J. 1999. Efficient mining of emerging patterns: discovering trends and differences. In Proc. of the 5th ACM SIGKDD int. conf. on Knowledge discovery and data mining, 43-52.

[2] Bayardo, Jr. R.J. 1998. Efficiently mining long patterns from databases. SIGMOD Rec. 27(2), 85-93.

[3] Agrawal, R. and Srikant, R. 1994. Fast algorithms for mining association rules. In Proc. of the 20th Int. Conf. on Very Large Data Bases (VLDB), 487-499.

[4] Mannila, H. and Toivonen, H. 1997. Levelwise Search and Borders of Theories in KnowledgeDiscovery. Data Min. Knowl. Discov. 1(3), 241-258.

[5] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H. and Verkamo, A.I. 1996. Fast discovery of association rules. In Advances in knowledge discovery and data mining, Fayyad,U.M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R. (Eds.). American Association for Artificial Intelligence, 307-328.

[6] Park, J.S., Chen, M.S. and Yu, P.S. 1995. An effective hash-based algorithm for mining association rules. In Proc. of the 1995 ACM SIGMOD int. conf. on Management of data (SIGMOD '95), 175-186.

[7] Brin, S., Motwani, R., Ullman, J.D. and Tsur, S. 1997. Dynamic itemset counting and implication rules for market basket data. In Proc. of the 1997 ACM SIGMOD int. conf. on Management of data (SIGMOD '97), 255-264.

[8] Zaki, M.J., Parthasarathy, S., Ogihara, M. and Li, W. 1997. New Algorithms for Fast Discovery of Association Rules. In Proc. of the 3rd Int. Conf. on Knowledge Disc. in Databases&Data Mining, 283-286.

[9] Gunopulos, D., Mannila, H. and Saluja, S. 1997. Discovering All Most Specific Sentences by Randomized Algorithms. In Proc. of the 6th Int. Conf. on Database Theory (ICDT '97), 215-229.

[10] Lin, D.I. and Kedem, Z.M. 1998. Pincer Search: A New Algorithm for Discovering the Maximum Frequent Set. In Proc. of the 6th Int. Conf. on Extending Database Technology: Advances in Database Technology (EDBT '98), 105-119.

[11] Mannila, H., Toivonen, H. and Verkamo, A. 1994. Improved methods for finding association rules. In Proc. AAAI Workshop on Knowledge Discovery.

[12] Rymon, R. 1992. Search Through Systematic Set Enumeration, In Proc. of the 3rd int. conf. on Principles of Knowledge Representation and Reasoning, 268-275.

[13] Dong, G., Zhang, X., Wong, L., and Li, J. 1999. CAEP: Classification by Aggregating Emerging Patterns. In Proc. of the 2nd int. Conf. on Discovery Science. Lecture Notes in Computer Science, 1721, 30-42.

[14] Li, J., Dong, G., and Ramamohanarao, K. 2000. Making Use of the Most Expressive Jumping Emerging Patterns for Classification. In Proc. of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Applications. Lecture Notes In Computer Science, 1805, 220-232.

[15] Li, J., Ramamohanarao, K. and Dong, G. 2000. The Space of Jumping Emerging Patterns and Its Incremental Maintenance Algorithms. In Proc. of the 17th Int. Conf. on Machine Learning (ICML '00), 551-558.

[16] Li,J., Dong,G. and Ramamohanarao, K. 2000. Instance-Based Classification by Emerging Patterns. In Proc. of the 4th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD '00), 191-200.

[17] Dong, G. and Li, J. 2005. Mining border description of emerging patterns from dataset pairs. Journal of Knowledge and Information Systems, 8(2), 178-202.

[18] Fan, H. 2004. Efficient mining of interesting emerging patterns and their effective use in classification, Ph.D. thesis, University of Melbourne.

[19] Ramamohanarao, K., and Bailey, J. 2003. Discovery of Emerging Patterns and Their Use in Classification. In Proc. of the 16th Australian Conference on Artificial Intelligence, 1-12

[20] Bailey, J., Manoukian, T. and Ramamohanarao, K. 2003. Classification using constrained emerging patterns. In Proc. of the 4th Int. Conf. on Web Age Information Management, 226-237.

[21] Bailey, J., Manoukian, T. and Ramamohanarao. K. 2002. Fast Algorithms For Mining Emerging Patterns. In Proc. of the 6th European Conf. on Principles of Data Mining and Knowledge Discovery, 39‐50.

[22] Han, J., Pei, J., and Yin, Y. 2000. Mining frequent patterns without candidate generation. In Proc. of the Int. Conf. on Management of Data, 1-12.

[23] Han, J., Pei, J., Yin, Y., and Mao, R. 2004. Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. Data Min. Knowl. Discov.,8(1), 53-87.

[24] Li, J., Ramamohanarao, K., and Dong, G. 2000. Emerging Patterns and Classification. In Proc. of the 6th Asian Computing Science Conference on Advances in Computing Science, 15-32.

[25] Fan, H. and Ramamohanarao, K. 2003. A Bayesian approach to use emerging patterns for classification. In Proc. of the 14th Australasian database conference - Volume 17 (ADC '03), 39-48.

[26] Fan, H. and Ramamohanarao, K. 2002. An Efficient Single-Scan Algorithm for Mining Essential Jumping Emerging Patterns for Classification. In Proc. of the 6th Pacific-Asia Conf. on Advances in Knowledge Discovery and Data Mining (PAKDD '02), 456-462.

[27] Ramamohanarao, K., Bailey, J. and Fan, H. 2005. Efficient Mining of Contrast Patterns and Their Applications to Classification. In Proc. of the 3rd Int. Conf. on Intelligent Sensing and Information Processing (ICISIP '05), IEEE Computer Society, 39-47.

[28] Stepinski, T.F., Ding, W. and Eick, C.F. 2011. Controlling patterns of geospatial phenomena. Journal Geoinformatica, 15(3), 399-416.

[29] Piao, M., Lee, J.B., Shon, H.S., Yun, U. and Ryu, K.H. 2011. Enumeration tree based emerging patterns mining by using two different supports. In Proc. of the 5th int. conf. on Convergence and hybrid information technology (ICHIT'11), 708-715.

[30] Tzanis, G., Kavakiotis, I. and Vlahavas, I.P. 2011. PolyA-iEP: A data mining method for the effective prediction of polyadenylation sites. Expert Syst. Appl, 12398-12408.

[31] Wang, L., Wang, Y., Jiang, T. and Gao, W. 2011. Instantly Telling What Happens in a Video Sequence Using Simple Features. In Proc. of the 24th IEEE int. Conf. on Computer Vision and Pattern Recognition (CVPR), 3257-3264.

[32] Chen, X. and Chen, J. 2011. Emerging Patterns and Classification Algorithms for DNA Sequence. Journal Of Software, 6(6), 985-992.

[33] Zhang, X., Dong, G. and Ramamohanarao, K. 2000. Exploring constraints to efficiently mine emerging patterns from large high-dimensional datasets. In Proc. of the 6th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD'00), 310-314.

[34] Zhang, X., Dong, G. and Ramamohanarao, K. 2000. Information-based classification by aggregating emerging patterns. In Proc. of the 2nd Int'l Conf. on Intelligent Data Engineering and Automated Learning (IDEAL'00), 175-188.

[35] Zhang,X., Guozhu Dong, and Kotagiri Ramamohanarao. 2001. Building Behaviour Knowledge Space to Make Classification Decision. In Proc. of the 5th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD '01), 488-494.

[36] Li, J., Dong, G., Ramamohanarao, K. and Wong, L. 2004. DeEPs: A new instance-based discovery and classification system. Machine Learning, 54(2), 99-124.

[37] Warnars, H.L.H.S. 2015. Mining Patterns with Attribute Oriented Inducation. In Proc. of The Int. Conf. on Database, Data Warehouse, Data Mining and Big Data (DDDMBD2015), 11-21.

[38] Han,J. and Fu, Y. 1995. Exploration of the power of attribute-oriented induction in data mining. in U. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy, eds. Advances in Knowledge Discovery and Data Mining, 399-421.

[39] Han, J., Cai, Y. and Cercone, N. 1992. Knowledge discovery in databases: An attribute-oriented approach. In Proc. of the 18th Int. Conf. Very Large Data Bases, 547-559.

[40] Li, J. and Wong, L. 2005. Structural geography of the space of emerging patterns. *Intell. Data Anal.* 9, 6 (Nov. 2005), 567-588.

[41] Warnars, H.L.H.S. 2012. Attribute Oriented Induction of High-level Emerging Patterns. In Proc. of Granular Computing Int. Conf. on Granular Computing (GrC).

[42] Warnars, H.L.H.S. 2014. Mining Frequent and Similar Patterns with Attribute Oriented Induction High Level Emerging Pattern (AOI-HEP) Data Mining Technique. Int, Journal Of Emerging Tech. in Computational and Applied Sciences (IJETCAS), 11(3), 266-276.

[43] Warnars, H.L.H.S. 2016. Using Attribute Oriented Induction High Level Emerging Pattern (AOI-HEP) to Mine Frequent Patterns. Int. Journal of Electrical and Computer Engineering (IJECE), 6(6), 3037-3046.

[44] Podraza, R. and Tomaszewski, K. 2005. KTDA: Emerging Patterns Based Data Analysis System. In Proceedings of XXI Fall Meeting of Polish Information Processing Society, 213—221.

[45] Boulesteix, AL., Tutz, G, and Strimmer K. 2003. A CART-based approach to discover emerging patterns in microarray data. Bioinformatics. 19(18),2465-2472.

[46] Fan, H. and Ramamohanarao, K. 2005. A weighting scheme based on emerging patterns for weighted support vector machines. In Proc. of IEEE Int. Conf. on Granular Computing (GrC), 435-440.

[47] Zhang, S., Ramamohanarao, K. and Bezdek, J.C. 2010. EP-based robust weighting scheme for fuzzy SVMs. In Proc. of the 21st Australasian Conf. on Database Technologies, 104, 123-132.

[48] Alhammady, H. And Ramamohanarao, K.. 2006. Using Emerging Patterns to Construct Weighted Decision Trees. IEEE Trans. Knowl. Data Eng, 18(7), 865-876.

[49] Ding, G., Wang, J. and Qin, K.. 2010. A visual word weighting scheme based on emerging itemsets for video annotation. Information Proces. Letters, 110(16), 692-696.

[50] Alhammady, H. And Ramamohanarao, K. 2004. Using Emerging Patterns and Decision Trees in Rare-Class Classification. In Proc. of the 4th IEEE Int. Conf. on Data Mining (ICDM), 315-318.

[51] Alhammady, H. And Ramamohanarao, K. 2004. The application of emerging patterns for improving the quality of rare-class classification. In Proc. of 8th Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD), 207–211.

[52] Alhammady, H. And Ramamohanarao, K. 2005. Expanding the Training Data Space Using Emerging Patterns and Genetic Methods. In proc. of SIAM Int. Data Mining Conf. (SDM), 481-485.

[53] Fan, H. and Ramamohanarao, K. 2006. Fast Discovery and the Generalization of Strong Jumping Emerging Patterns for Building Compact and Accurate Classifiers. IEEE Trans. on Knowl. and Data Eng, 18(6), 721-737.

[54] Appice, A., Ceci, M., Malgieri, C., and Malerba, D. 2007. Discovering Relational Emerging Patterns. In Proceedings of the 10th Congress of the Italian Association For Artificial intelligence on AI*IA 2007: Artificial intelligence and Human-Oriented Computing (Rome, Italy, September 10 - 13, 2007). R. Basili and M. T. Pazienza, Eds. Lecture Notes In Artificial Intelligence. Springer-Verlag, Berlin, Heidelberg, 206-217.

[55] Ceci, M., Appice, A., and Malerba, D. 2007. Discovering Emerging Patterns in Spatial Databases: A Multi-relational Approach. In *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases* (Warsaw, Poland, September 17 - 21, 2007). J. N. Kok, J. Koronacki, R. Lopez De Mantaras, S. Matwin, D. Mladenič, and A. Skowron, Eds. Lecture Notes In Artificial Intelligence. Springer-Verlag, Berlin, Heidelberg, 390-397.

[56] Ceci, M., Appice, A., and Malerba, D. 2008. Emerging Pattern Based Classification in Relational Data Mining. In *Proceedings of the 19th international Conference on Database and Expert Systems Applications* (Turin, Italy, September 01 - 05, 2008). S. S. Bhowmick, J. Küng, and R. Wagner, Eds. Lecture Notes In Computer Science. Springer-Verlag, Berlin, Heidelberg, 283-296.

[57] Knobbe, A., Blockeel, H., Siebes, A., Van der Wallen, D.M.G., *Multi-Relational Data Mining*, In Proceedings of Benelearn '99, 1999

[58] Knobbe, A., *Multi-Relational Data Mining*, Ph.D. Dissertation, Kiminkii, the Netherlands, 2004

[59] Džeroski, S. 2003. Multi-relational data mining: an introduction. *SIGKDD Explor. Newsl.* 5, 1 (Jul. 2003), 1-16.