Proceedings of the International Multiconference on Computer Science and Information Technology pp. 73–79
ISBN 978-83-60810-27-9
ISSN 1896-7094

Coevolutionary Algorithm For Rule Induction

Paweł B. Myszkowski
Wrocław University of Technology,
Wyb. Wyspiańskiego 27, 51-370 Wrocław, Poland
Email: pawel.myszkowski@pwr.wroc.pl

***Abstract*—This paper describes our last research results in the field of evolutionary algorithms for rule extraction applied to classification (and image annotation). We focus on the data mining classification task and we propose evolutionary algorithm for rule extraction. Presented approach is based on binary classical genetic algorithm with representation of 'if-then' rules and we propose two specialized genetic operators. We want to show that some search space reduction techniques make possible to get solution comparable to others from literature. To present our method ability of discovering the set of rules with high F-score we tested our approach on four benchmark datasets and ImageCLEF competition dataset.**

Keywords: **data mining, rule extraction, evolutionary algorithms, image annotation**

I. Introduction

The size of datasets is growing constantly and as cannot be analysed it by human being so we use automating process, so-called Knowledge Discovery in Databases (KDD) which is a part of Machine Learning domain. The most interesting, from this paper point of view, is its one stage – data mining (DM). The data mining is an interdisciplinary field and its essence is knowledge acquisition from large amount of data. As our data might contain useful hidden and implicit the knowledge, the extracted knowledge can be successfully used in very important real-world domains such as image annotation.

As Evolutionary Algorithms (EA) are metaheuristics that search the solution search space and can be easily applied for data mining tasks, such as clustering, prediction, classification and rule induction (paper [8] is a great survey of EA applications to KDD). Rule discovering (or rule induction) is most studied data mining task and its main goal is to build model of given data that describes it with possible best accuracy. Such model can be based on intuitive 'if-then' rules where: if-part (antecedent) attribute conditions and then-part (consequent) contain predicted class value (label). The classifier quality (accuracy of the gained model) can be tested on unseen data and measured by prediction error value.

EA is group of metaheuristics inspired by nature (Darwinian evolution theory) where the fittest individuals have better chance to survive and EA is widely used in data mining tasks (e.g. [2][3][5][15][20]). EA codes the problem solution as an individual and operates its features with aid of genetic operators (usually by mutation and crossover). Quality of individual is given as fitness function and its better value gives the higher probability of getting an offspring. In rule discovering task, the main motivation is to discover the rules with high predictive accuracy value. In literature we can find some approaches based on natural evolution that extends simple EA (e.g. classical genetic algorithm [9][12]) by some additional elements. For instance, commonly used if-then form of rule can be a single individual (Michigan approach) or included in ruleset form of individual (Pittsburgh approach). Mostly, there is used the Pittsburgh approach as it takes into consideration a rule interaction and is more natural and intuitive.

Interesting EA based method, so-called Genetic Programming (GP) can be found in [3], where individual is represented by logic tree that corresponds to logical expression such as rule: "If $attrib_0$>5 and ($attrib_3$=2,5 or $atrrib_2$<1,3) then $class_3$" to describe attributes' conditions of given class in chest pain diagnosis. The rule is presented as decision tree, where internal nodes are operators and leaf nodes represents attributes and corresponding condition values.

The multipopulation EA is based on the natural phenomena of coevolution (CoEvolutionary Algorithm, CoEA), where fitness function evaluation of few populations runs effectively in parallel way. Such method is proposed in [15], where individuals are selected from a few populations by cooperative selection pressure by choosing the best, previous fitness or classical selection method. There is used a collaboration pool size (from each population a group of the candidate individuals is selected) or a collaboration credit assignment (based on a selection of n individuals, where its optimistic, the average or pessimistic version is used). Another CoEA (presented in [20]) operates on populations that correspond to ruleset of n rules (where n is a parameter) linked by token competition as a form of the niching method. Paper [17] describes approach that takes into consideration distributed genetic algorithm for rule extraction task, where is presented positive influence of dynamic data partitioning distribution model to classifier accuracy.

There is also another strong trend in EA applications in DM – usage of specialized genetic operators. In Pittsburgh approach to classification task it is very important that individual representation consists of a complete classifier where rules cooperate and recombination operator causing its separation may make worse its classification accuracy. In [20] above problem is discussed and there is proposed a symbiotic combination operator which is a kind of heuristic that ana-

This work is partially financed form the Ministry of Science and Higher Education Republic of Poland resources in 2008-2010 years as a Poland-Singapore joint research project 65/N-SINGAPORE/2007/0.

lyzes results of changes in newly created individual. Another type of genetic operator specialization can be the usage of some hill-climbing algorithms to improve individuals (as candidate solution) by making "minor" modifications: if it causes fitter individual, given change is accepted. As a matter of fact, this causes the Baldwin effect [12]. Also, in evolution process ruleset can be modified in pruning procedure (e.g. [5]) - is optimized by removing unused/invalid attribute condition according to information gain measure value and/or examines results by some small changes in the ruleset. Also, we can find hybridization as a quite strong trend, where main motivation of such propositions is to build approach and link advantages of connected methods.

The remainder of this paper is organized as follows. Details of problem definition and our approach for rule discovering task is presented in section 2. Research methodology, used benchmark dataset and results of experiments are presented in section 3. Finally, section 4 presents conclusions and future research directions.

II. CAREX: Coevolutionary Algorithm for Rule EXtraction

Proposed method uses standard EA schema and starts a learning process with initial population (usually created randomly) and next individuals of current population are evaluated: each individual receives a fitness function value that corresponds to quality of proposition of given problem solution. In next step EA checks if stop conditions are not met: usually it is limit of generations and the best individual fitness value is acceptable (success). If stop criteria is not met EA runs the selection procedure that defines a seed of the new generation; next it is a communication between individuals (by crossover operator) and the independent trial (by mutation operator). The whole process repeats until some stopping condition is met. The crucial issue in evolutionary based method is the definition of individual representation schema, genetic operators (mutation and crossover) and evaluation function form, that give information about the individual fitness function value. In this section above elements are described.

A. Representation schema

In CAREX approach we decided to construct individual representation schema as simple as possible to get reduction of solution search space size. This is gained by coding of arguments value in binary representation (usually 2x8 bits per arguments) and only two logical operators: IN and NOT_IN. This methodology makes possible to use simple genetic algorithm, as we wanted to show that there is no need of EA extension to get solution comparable to others based on literature. Also we shown that there is possibility to build a specialized genetic operators.

We considered Michigan and Pittsburgh [12] approaches and finally CAREX system implements both, but in our research we use only the Pittsburgh model to keep all rules interactions in one individual. Therefore individual is represented as the set of rules (*ruleset*) that can assign instance to one class (if the task is to find completely one class description, so-called one-class-model) or to get model of all classes presented in dataset the individual consists of rules connected to class identifiers (all-class-model) as follows:

$$RuleSet := \{Rule_0, \ldots, Rule_n\}$$

Each rule is represented as the set of commonly used if-then type rules IF <*attributes_conditions*> THEN <*class_identifier*> as follows:

$$Rule_i := IF\ A_0 \text{ and} \ldots \text{and } A_n\ THEN \text{ class}_j$$

Where $class_j$ symbol represents given class identifier, and $A_i, i \in \{0, \ldots, n\}$ represents a numeric range for a condition for *i*-th attribute as follows:

$$A_i := attribute_i\, operator\, (a,b)$$

where bellow *operator* can be:

$$\text{IN}(a,b) \rightarrow a < attribute_i < b \text{ or}$$

$$\text{NOT\_IN}(a,b) \rightarrow attribute_i < a \text{ or } attribute_i > b$$

where $a<b$ and $a,b \in \mathbb{R}$. Above representation is strictly based on conditions combination for selected attributes. We decided to use only two operators to keep individual representation as simple as possible. For instance, rule is:

$$IF\ attribute_0 \text{ IN}(0.1, 0.5) \text{ and}$$
$$attribute_2 \text{ NOT\_IN}(1.0, 1.2)\ THEN\ class_2$$

Above rule describes all instances with values from range <0,1; 0,5> for $attribute_0$. Another condition takes into consideration $attribute_2$, where its value cannot be in range <1,0; 1,2>. Data described by conjunction of conditions are proposed to label with $class_2$.

In CAREX we decided to use binary vector representation thus we are allowed to use classical binary genetic operators to manipulate individual's particles. To avoid a drastic change of attribute value we use a Gray code. Also each attribute is extended by one enabled/disabled bit. Before CAREX starts, the dataset is preprocessed: instances are analyzed to recognize domains for all attributes. Then, each attribute domain is mapped into binary vector. That allows to keep each individual valid and there is no need to waste extra CPU time for repairing or removing invalid attribute values.

In proposed representation we use only "AND" logical operator, therefore EA to describe some set of instances as two separate conditions using ruleset as two connected rules. Indeed, the relation between these rules is logical disjunction, and indeed there exist a sort of rules coevolution phenomena.

B. Fitness Function

Generally, in EA, fitness function evaluation is very critical issue. Its definition decides about shape of solution landscape and must be defined very carefully. As rule extraction problem corresponds to data mining and our individual is a ruleset we can use commonly used classifier measure. In

classification the main goal is prediction of the value $c_i \in C$ (class) analyzing values of attributes x_i of given instance $x_i = \{x_i^0, \ldots, x_i^n\}$ where $x_i^n \in X$ defines solution landscape. Thus classification task is based on explore set of $(x_1, c_1), \ldots, (x_n, c_m)$ to build model $m(x_i): X \to C$ that labels unseen instance $x_k \in X$. Evaluation of rule is connected to its quality as classifier. In such context of data mining domain, the terms true positives (*tp*), true negatives (*tn*), false positives (*fp*) and false negatives (*fn*) we can define *recall* :

$$recall = \frac{tp}{tp+fp} \tag{1}$$

and *precison* measure as follows:

$$precision = \frac{tp}{tp+fn} \tag{2}$$

The *recall* value tells only if given rule labels instances in proper way, *precision* informs if rule covers all labeled data by proper class identifier. Above formulas say a lot about rule classification quality but there are two separate values. Although EA should use only one value to evaluate rule in literature (e.g. [3]) we can find some combinations of these values. However, we decided to use measure based on modified van Rijsbergen's effectiveness measure (F*score*) [18], than can be used also in data mining, as it combines *precision* and *recall*:

$$Fscore = \frac{1}{\frac{\alpha}{precision} + \frac{1-\alpha}{recall}} \tag{3}$$

where α can give a predominance of one of two elements, but we established its value on 0,5 to keep two elements equal. If *Fscore* value is near 1 means that evaluated rule has high quality as it corresponds to maximizing problem. Bellow *Fscore* measure is very useful as fitness function form, but for comparison of gained results in literature is used other measure of predictive *accuracy*, as a rule quality measure, defined as follows:

$$accuracy = \frac{tp+tn}{tp+fp+fn+tn} \tag{4}$$

where *tp/tn/fp/fn* correspond respectively to true positive/true negative/false positive/false negative to denote accuracy of classification in given set of instances.

We do not use accuracy formula as fitness function because of more practical features of *Fscore.* As individual is a ruleset the fitness function value is calculated as the average value of *Fscore* of all existing in dataset classes. Such form of fitness function occurs some distortions specially when dataset is dominated by one class and others have small representation. In dominated datasets we can use stratified crossover mechanism to reduce such problem.

C. Genetic operators

Our individual is represented by binary vector which can be operated by classic binary operators in simple way. Random modification of selected bit works as mutation (***SM***, Simple Mutation) and inserts new information into chromosome. To deliver communication in population in CAREX we developed one-point crossover (***OX***) that links two individuals to build the new one as combination of their genes. As all individuals have the same size, there is no situation of invalid individual creation using the random cut position. The high efficiency of basic set of genetic operators (*SM* and *OX*) in CAREX encouraged us to construct some specialized operators. We developed and tested two: Directed Mutation (*DM*) and Best Class Crossover (*BCX*).

Directed Mutation (DM)

This operator is not semi blind as simple mutation *(SM)* and it tries to direct evolution process by increasing the mutation probability if given rule has low quality (fitness function value is low). Given $Rule_i$ mutation probability is given according to below formula:

$$P_m' = P_m * (2 - Fit^{class_i}(Rule_i)) \tag{5}$$

where P_m is constant mutation probability given to whole *EA*. The $Fit^{class_i}(Rule_i)$ formula part gives a value of fitness value of given $class_i$ classification accuracy. The motivation is to change rules in class that has worst quality value (near 0,0), but it is almost absent when value is near 1,0 (the best quality). Of course, such directed mutation selects rules to modification but if such direction is too hard it could stuck the whole learning process in local optima. To avoid such situation we establish an increasing parameter as maximal value to double P_m value.

Best Class Crossover (BCX)

The standard crossover operator (such as *OX*) does not take into consideration specific problem knowledge. It just randomly cuts individuals and randomly generates a new individual. Our main motivation was to build a new crossover operator that uses specific domain knowledge and links two individuals more effectively. We inspired slightly the *BCX* operator (presented in [13] which, in its base form, is used for graph coloring problem, we developed the other version of *BCX* (Best Class Crossover), which is presented as pseudo code on Błąd: Nie znaleziono źródła odwołania.

The BCX operator takes two RuleSet P_1 and P_2 as parental individuals and returns the newly generated as result. The offspring is compilation of two parental individuals based on analyse which a parental *RuleSet* has better accuracy of given class description c_i – its rules are included in newly generated offspring individual. Such operation is repeated for each class independently.

The operator is specific form of communication between individuals where each of them gives the best part of selected rules and it is very useful, but its frequent usage is rather

```
function crossover_BCX ( RuleSet P1,
                         RuleSet P2 )
  begin
  RuleSet  Off := Ø
     foreach class ci
       begin
             if  (Fit^ci(P1^ci) > Fit^ci(P2^ci))
                 then  Off := Off ∪ P1^ci
                 else  Off := Off ∪ P2^ci
       end
  return  Off
  end
```

Fig 1. BCX operator pseudocode

risky. As it is determinate, it can cause loss of diversity in population and premature convergence indeed. Such operator we use not more than P_x<0,1.

D. CAREX programming platform speedup

Our experiments needed programming platform and we decided to use Java language. To make experiments less time consuming we developed:

- data buffering, all data are stored in RAM memory not in hard drive – it makes data query more effective and less time consuming,
- data indexing based on binary search that gives us more effectively access to our data,
- data sequence covering based on attributes condition cascade techniques that give evaluation function of each rule less time consuming,
- evaluation cache, that stores in RAM memory rules that were already evaluated and in evolution process are without any changes. Thanks this mechanism time for evaluations is significantly reduced.

Such low level programming techniques (not connected to EA conception directly) make experiments less time consuming. It is worth mentioning because of stochastic character of EA, where experiments must be repeated many times.

III. Computational experiments and results

Evaluation of learning method is important to compare results against other methods. This can be done experimentally. We developed in Java a research environment that supports learning and rule validating process. To evaluate predictive classification ability of developed model we split data into train and test data using commonly used crossvalidation method.

First, there are used train data to generate *RuleSet* by CAREX to get possible high accuracy (*Fscore* is used as fitness function). Learning process based on evolution runs using selection, mutation and crossover operators. For evaluation accuracy of gained rules there is used train dataset, but when evolution process is finished test dataset is used to validate predictive accuracy of generated *RuleSet*. To minimalize a random elements influence on our research we use 5x2 crossvalidation (according to research presented in [7]). Also to reduce influence of stochastic aspect of EA to our research the whole evolution process is repeated 10 times to each crossvalidation fold.

A. Used datasets

To show CAREX application we decided to use benchmark data set from UCI data Repository [21]. From set of 187 datasets we selected four connected to important real-world domains such as biology, medicine, chemical industry, food industry or health care, as follows:

- **iris**: 150 instances, 4 attributes, 3 classes (distribution: 50/50/50),
- (pima indians) **diabetes**: 768 inst., 8 attributes, 2 classes (distrib.: 500/246),
- **glass**: 214 instances, 9 attributes, 6 classes (distribution: 70/17/76/13/9/29)
- **wine**: 178 instances, 13 attributes, 3 classes (distribution: 59/71/48)

First dataset (Fisher's iris flower) is benchmark for each new classification algorithm. The diabetes dataset includes medical records of medical review if the patient shows signs of diabetes according to the World Health Organization criteria. The third dataset consists of glass identification in chemical industry data. The last one (wine) contains results of chemical analysis of wines grown in the same region in Italy.

B. Experimental CAREX

We experimented to establish CAREX algorithm optimal parameters values. The most influential parameter is the probability of mutation (see its influence on average predictive accuracy presented on Fig 1). Our experiments showed that CAREX results are the most stable and the best in P_m=0,01. Probability of crossover P_x=0,2 but our first research result showed that crossover has minor influence on CAREX results.

Another very important CAREX issue is population size and number of generations. We developed several experiments series to establish its optimal values. Results of such experiments are showed in Fig 2, where influence of number of births (as product of population size and number of generations) to average accuracy of developed model in iris dataset is presented. We used population size and generations number from set {10,20,50,100,200 and 1000}. The Fig 3 shows that the optimal value of births is between 10 and 20 thousands, which is connected to population size 100 and 200-400 generations.

The higher values of births make CAREX less effective – it is more time consuming learning process and we do not have the significant increase of accuracy value.

Basically, we tested two types of selection methods: tournament selection and roulette wheel selection. The best results were achieved in tournament selection, where individuals are selected without replacement to the pool size of 2 or 10 individuals. The roulette wheel method causes smaller selection pressure and makes learning process less effective. In our experiments we do not use elitism parameter.

C. Effectiveness of CAREX

This part of paper presents experiments connected to population size and genetic operators usage influence on CAREX effectiveness. Evaluation of CAREX approach in the context of four benchmark dataset is given. Used values of EA parameters are presented in Table 1.

Table 1. CAREX parameters used in experiments

parameter name	IRIS	DIABETES	GLASS	WINE
Population size	200	20	200	100
Number of generations	200	200	500	200
Prob. of crossover [OX]	0,2	0,2	0,2	0,2
Prob. of mutation [DM]	0,01	0,01	0,01	0,01
Selection method	10	2	10	10
Number of rules	5	10	30	15

In CAREX an individual is represented by set of rules and this number strictly depends on used dataset: usually we use about 3-5 rules per class. As different datasets have various CAREX parameters requirements are presented in Table 1. The population size in our approach usually equals to 100-500 individuals. Number of generations gives evolution process the "time" to work out a solution but we experimentally tested that there is no need to extend it to not more than 1000 generations. Another important aspect of EA is selection – we experimentally developed a tournament selection of 2 or 10 individuals. A mutation operator (*Pm*) rate is experimentally set to 0,01. Stop condition is met if individual that has *Fsc* equals to 1,0 (success) or number of generation is exceed (fail).

Corresponding to literature we compare results of CAREX with other methods (see Table 2) - we selected EA based methods (e.g. CORE [20], CCE [15], ESIA [11]) and classical C4.5 [16] algorithm. For each dataset we presented results of each selected method (as it is given in literature): the average predictive accuracy, standard deviation value and maximal value of observed accuracy. Unfortunately, not for all dataset the results are given in the literature.

We can observe that CAREX achieves the best accuracy for **iris** dataset. Also comparable results gain CORE method and EMA-AIS. It is worth to mention that difference is very small (near to statistical error value). For **diabetes** dataset we can see that CAREX does not achieve the best accuracy (but comparable to classical C4.5 algorithm). It is outperformed by CORE and EMA-AIS method. It is also worth mentioning that EMA in its basic form gives comparable results to CAREX, while EMA-AIS is the hybrid of EMA with AIS and gives potentially better solution. Comparison of predictive accuracy value in **glass** dataset to other methods shows that CAREX results can successfully compete with other methods. Unfortunately, stochastic element of CAREX causes relatively large value of standard deviations (8,75%). That value is very puzzling, because ESIA approach uses 5-fold cross-validation and standard deviation equals to 0,03%. Dataset **wine** results presented in the above table show that given dataset is no problematic for CAREX. We compared our results with two other approaches, and CAREX outperforms classic C4.5 and gives results comparable to SEA method.

C. ImageCLEF Photo Annotation competition dataset – first results

In previous section we presented CAREX method applied to benchmark UCL Repository datasets. Improved high CAREX efficiency, we applied it to practical problem. We

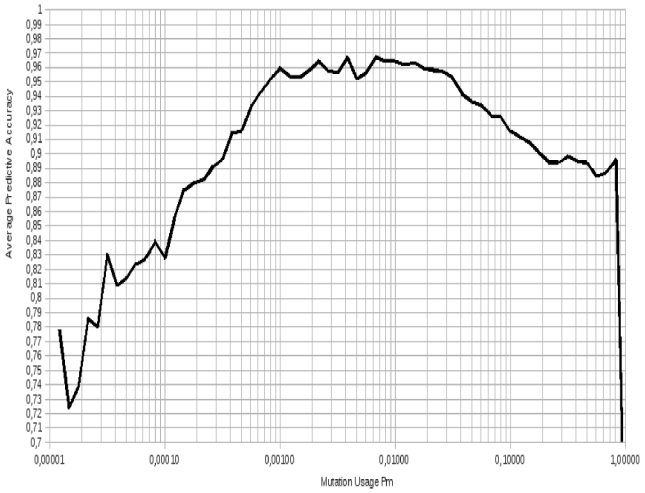

Fig 2. CAREX mutation prob. value' influence on average predictive accuracy [iris dataset]

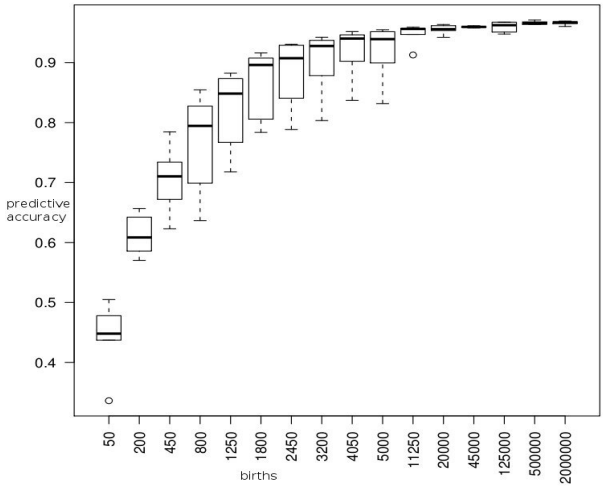

Fig 3. CAREX - number of births and its influence on average predictive accuracy [iris dataset]

Table 2. Performance comparison for the benchmark dataset.
Average predicative accuracy [%], standard deviation and best value of accuracy are given [%].

	IRIS	DIABETES	GLASS	WINE
CAREX	**96,67±1,5 (99,1)**	73,89±1,0 (76,43)	**75,49±8,75 (83,54)**	**93,93±3,68 (100)**
CORE [20]	**96,61±2,35 (100)**	**75,34±2,3 (80,15)**	n.u.	n.u.
C4.5 [16]	93,67±3,73 (100)	73,13±2,5 (77,39)	67,72±12,27 (?)	89,03±7,55 (?)
CCE [15]	95,40±3,28 (100)	n.u.	n.u.	n.u.
ESIA [11]	95,33±3,0 (?)	70,18±0,21 (?)	72,43±0,03 (?)	n.u.
EMA [2]*	94,59±3,9 (100)	73,23±2,13 (77,08)	n.u.	n.u.
EMA-AIS [2]*	**97,02±0,83 (100)**	**75,23±1,4 (77,6)**	n.u.	n.u.
SEA [10]	95,57±? (?)	n.u.	0,68±? (?)	**94,59±7,55 (?)**

decided to generate rules in photo annotation problem – and we used benchmark dataset from ImageCLEF conquest [14]. Automatic Image/photo annotation problem [4] needs method for multiclass labeling images, as problem concerns a number of classes and large size of datasets. There is many approaches in literature based on decision trees, suport vector machine (SVM) and others (survey can be found in [1]).

The ImageCLEF conquest dataset consists of 8000 images labeled by 93 classes (words). We do not use any segmentation methods and we use 18 global image Tamura [19] features such as coarseness, contrast or directionality. We run CAREX method using 10 individuals to run 100 generations to get one word annotation. The example of such run is

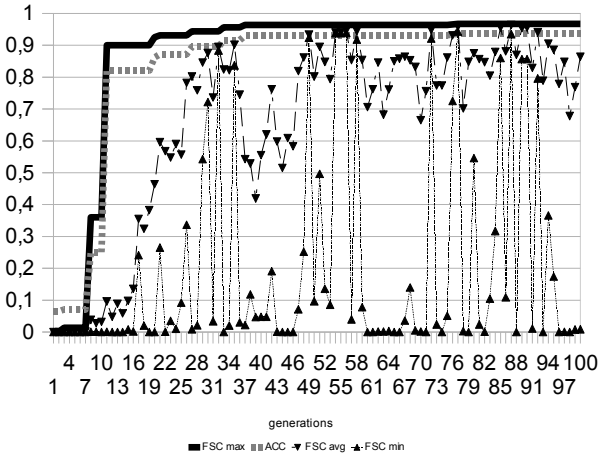

Fig 4. Example of CAREX run of Photo ImageCLEF word 'Neutral_Illumination'. CAREX parameters: 2 Rules, popsize=10, generations=100, P_m=0.01, P_x=0.2

presented on Fig 4, where is shown the best *Fscore*, average *Fscore*, worst *Fscore* and best Accuracy in given evolution process. Evolutionary Algorithm searches the whole solution space effectively and even so small population (only 10 individuals) makes possible to get optimal solution.

Some initial experiments were done and effectiveness of presented CAREX method is very promising. First results were presented in Table 3, where the best of 10 word were shown. The average *Fscore* for 10 chosen words equals to 74,5%. However, these words are frequent and the average F-score value in dataset equals 26%. Example of 2 rules

Table 3. CAREX results for 10 best words for ImageCLEF images dataset

word	records	Fscore	Precision	Recall	Accuracy
Visual_Arts	3346	0,590	0,419	1,000	0,420
No_Visual_Time	3271	0,590	0,429	0,946	0,462
Cute	3909	0,657	0,490	0,998	0,491
Outdoor	4172	0,697	0,560	0,923	0,581
Day	4198	0,710	0,570	0,941	0,597
Natural	4593	0,730	0,575	0,999	0,576
No_Blur	5240	0,792	0,656	0,999	0,656
No_Persons	5467	0,812	0,684	1,000	0,684
No_Visual_Season	6646	0,908	0,831	1,000	0,831
Neutral_Illumination	7483	0,968	0,939	1,000	0,938
	average	0,745			

generated on word 'Neutral_Illumination' with very high value of *Fscore*:

```
IF       a4 IN<58,128;341,378>
    AND  a7 NOT_IN<170,012;266,089>
    AND  a8 NOT_IN<291,938;312,497>
    AND  a13 NOT_IN<169,923;265,895>
    AND  a14 IN<178,128;345,282>
    AND  a15 IN<19,201;218,386>
THEN class = 'Neutral_Illumination'
  [Rec=0,013 Prec=0,981 Acc=0,077 Fsc=0,027]
IF       a7 NOT_IN<586,874;603,571>
THEN class = 'Neutral_Illumination'
  [Rec=1,000 Prec=0,935 Acc=0,935 Fsc=0,967]
```

The CAREX in the first results is very promising but also needs some extra ImageCLEF experiments connected to specialized genetic operators, another set of features (not only 18 Tamura features), segmentation influence to CAREX effectiveness and time consuming EA run as extra

tests. Another positive aspect of CAREX usage is very intuitive rule representation that allows to understand the annotation description by the human.

IV. Conclusions and further work

The presented method CAREX is based on EA and simple binary solution representation. We wanted to show that binary based EA can be effective in searching a large search space in data mining tasks. We presented results of performance based on four benchmark databases. The CAREX results are compared to other methods and show that CAREX is able to compete successfully in rule induction/classification task. Also, in our opinion CAREX has great potential in this area of research. However, experiments results showed some problems, especially connected to stochastic aspects of presented approach – in our opinion the standard deviation value of results is too high to be accepted in real-world applications. We presented first experiment results on real world of image annotation problem using the ImageCLEF dataset. Results of developed experiments encourage to continue work on CAREX.

We plan further research into two main directions. First, fitness function modification by adding some niching methods to make it more competitive. Other way we see in specialization of genetic operators, including some ruleset improving methods such as hill climbing, local search type or simple pruning rules procedure and discovering potential labeling conflicts in the whole ruleset.

V. References

[1] Alham N. K., Li M., Hammoud S., Qi H. "Evaluating Machine Learning Techniques for Automatic Image Annotations" Proc. of the 6th Inter. Conf. on Fuzzy Syst. and Know. Discovery. vol 07, pp: 245-249, 2009

[2] Ang J. H., Tan K. C., Mamun A. A., *An evolutionary memetic algorithm for rule extraction*, Expert Systems with Applications 37, pp.1302-1315, 2010.

[3] Bojarczuk C., Lopes H., Freitas A., *Genetic programming for knowledge discovery in chest pain diagnosis,* IEEE Eng. Med.Mag 2000:19(4):38-44, 2000.

[4] Broda B., Kwasnicka H., Paradowski M., Stanek M.: MAGMA - efficient method for image annotation in low dimensional feature space based on Multivariate Gaussian Models. IMCSIT 2009 proceedings 131-138, 2009.

[5] Cattral R., Oppacher F., Graham Lee K. J., *Techniques for Evolutionary Rule Discovery in Data Mining*, IEEE Congress on Evolution. Comp., Norway 2009.

[6] Carneiro G., Chan A.B., Moreno P.J., Vasconcelos N., "Supervised Learning of Semantic Classes for Image Annotation and Retrieval," IEEE Trans. on Pattern Analysis and Machine Intelligence 29(3), pp. 394-410, 2007

[7] Dietterich T. G., *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithm*, Neural Comp. 10, pp.1895–1923, 1998.

[8] Freitas A. A. *A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery,* Advantages in Evolutionary Computing: theory and applications, pp.819-845, Spinger-Verlag NY, 2003.

[9] Goldberg D., *Genetic algorithms in search, optimization and machine learning*, Addision-Wes., 1989.

[10] Halavati R., Souraki S.B., Esfandiar P., Lotfi S., *Rule Based Classifier Generation using Symbiotic Evolutionary Algorithm*, ICTAI, vol. 1, pp.458-464, 19th IEEE Inter. Conf. on Tools with AI, 2007.

[11] Liu J. J., Kwok J. T. *An extended genetic rule induction algorithm*, Proceedings of the Congress on Evolutionary Computation (CEC), pp.458-463, La Jolla, California (USA), 2000.

[12] Michalewicz Z., *Genetic Algorithms+Data Structures =Evolution Programs,* Springer-Verlag, 1994.

[13] Myszkowski P. B., *Solving Scheduling Problems by Evolutionary Algorithms for Graph Coloring Problem*. Metaheuristics for Scheduling in Industrial and Manufacturing App, pp.145-167, Springer -Verlag 2008.

[14] ImageCLEF Photo Annotation competition dataset: http://www.imageclef.org/2010/PhotoAnnotation

[15] Stoen C., *Various Collaborator Selection Pressures for Cooperative Coevolution for Classification*, International Conference of Artificial Intelligence and Digital Communications, AIDC 2006.

[16] Quinlan J. R. *Bagging, Boosting, and C4.5*, Proc. of the 13th National Conf. on AI, pp. 725-730, 1996.

[17] Rodriguez M., Escalante D.M., Peregrin A., *Efficient Distributed Genetic Algorithm for Rule Extraction*, Applied Soft Computing. (accepted, Jan 13th, 2010).

[18] van Rijsbergen, C. J. (1979). *Information Retrieval* (2nd ed.). Butterworth. 1979.

[19] Tamura H., Mori S., Yamawaki T. "Textrual Features Corrensponding to Visual Perception", IEEE Transactions on Systems, MAN and Cybernetics 8(6), pp.460-473, 1978.

[20] Tan K. C., Yu Q., Ang J. H., *A coevolutionary algorithm for rules discovery in data mining*, Inter. Jour. of Systems Science, Vol.37, No.12,pp.835-864, 2006.

[21] UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/)