jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2010, VOL. 56, NO. 4, PP. 427–432
Manuscript received April 30, 2010; revised July 2010. DOI: 10.2478/v10177-010-0058-4

Mining Pharmacy Database Using Evolutionary Genetic Algorithm

Mourad Ykhlef and Hebah ElGibreen

***Abstract*—Medication management is an important process in pharmacy field. Prescribing errors occur upstream in the process, and their effects can be perpetuated in subsequent steps. Prescription errors are an important issue for which conflicts with another prescribed medicine could cause severe harm for a patient. In addition, due to the shortage of pharmacists and to contain the cost of healthcare delivery, time is also an important issue. Former knowledge of prescriptions can reduce the errors, and discovery of such knowledge requires data mining techniques, such as Sequential Pattern. Moreover, Evolutionary Algorithms, such as Genetic Algorithm (GA), can find good rules in short time, thus it can be used to discover the Sequential Patterns in Pharmacy Database. In this paper GA is used to assess patient prescriptions based on former knowledge of series of prescriptions in order to extract sequenced patterns and predict unusual activities to reduce errors in timely manner.**

***Keywords*—Data mining, evolutionary algorithms, genetic algorithm, pharmacy database, sequential patterns.**

I. Introduction

MEDICATION MANAGEMENT is an important process especially in pharmacy field. In this process, prescribing errors might occur and place patients at risk of adverse drug events. Prescribing errors are a particular concern for which conflicting with another prescribed medicine could cause severe harm for a patient. The knowledge of prescriptions can reduce the risk of harm to patients from prescribing errors. Extracting these knowledge will give the pharmacist general awareness of prescriptions that will alert him when an unusual activity occur in order to check again with the doctor.

In Pharmacy Database, it is important to analyze such large data because they may contain new knowledge. The discovery of such knowledge requires Sequential Pattern mining, which is one of data mining technologies. It extracts patterns that appear more frequently than a user-specified minimum support, while maintaining their item occurrence order.

Due to the shortage of pharmacists there is urgency for efficiency improvement in pharmacy operations. This improvement is also important to contain the cost of healthcare delivery. Mining Sequential Pattern algorithms takes a long time to find the rules especially when they are applied on large databases. On the other hand, the evolutionary algorithms can find a good Sequential Pattern rules within a short time. Consequently, Genetic Algorithm (GA), which is an evolutionary algorithm, can be used to discover Sequential Pattern rules in a short time. GA is a general purpose search algorithm which uses principles inspired by natural genetic populations to evolve solutions to problems.

M. Ykhlef and H. ElGibreen are with King Saud University, College of Computer and Information Sciences, Information System Department, Kingdom of Saudi Arabia (e-mails: ykhlef@ksu.edu.sa, HJibreen@ksu.edu.sa).

Related Work: Agrawal and Srikant introduced the Sequential Pattern mining problem in [1], [2], while Genetic Algorithm has been introduced in [3]–[5]. GA has many representations for the chromosome, presented in [6]–[8], and we studied these representations and chose the best one based on David advice [4]. Additionally, there are many fitness function measures, presented in [9]–[11], and Piatesky-Shapiro [3] suggested some principles and property to choose the most appropriate measure depending on the problem. We applied his principles to choose the most appropriate measure for the proposed algorithm. Kaya and Alhajj [12] proposed a novel multi-objective Genetic Algorithm method for optimizing quantitative fuzzy Sequential Patterns; they applied GA on fuzzy sequential patterns using multi-objective measure. In [13], the writers introduced the risk of prescribing errors. They applied process-improvement initiatives to reduce the risk of harm to children resulting from prescribing errors and they focused on prescriber education and behavior modification. In this paper, Genetic Algorithm is combined with Sequential Pattern mining using the sequential interestingness measure and then applied to Pharmacy Database, in order to reduce the risk of harm to patients resulting from prescription errors.

The rest of the paper is organized as follows. Sequential Patterns and Genetic Algorithm are defined in section II and III. The proposed approach of applying GA to Sequential Patterns, in Pharmacy Database, is described in Section IV. Experimental results are reported in Section V. At the end, conclusion with some future works is presented in section VI.

II. Sequential Patterns

Sequential Pattern mining addresses the problem of discovering the existent maximal frequent sequences in a given database. The problem was first introduced by Agrawal and Srikant [1], [2], where the basic concept involved in pattern detection has been established. It seeks similar patterns in data transaction; this approach is useful when the data to be mined has some sequential nature to deal with databases that have time-series characteristics, i.e. when each piece of data is an ordered set of elements [14]. For example, it can be said, 60% of patient who take medicine X will take medicine Y afterward, regardless of the time gap.

Given a Pharmacy Database, where each transaction includes a patient ID, prescription time and its medicine, as in Table.1, Sequential Pattern can be defined as follows.

Definition 1: Let $I = \{x_1 \dots x_n\}$ be a set of items. An itemset is a non-empty subset of items, and an itemset with k items is called k-itemset. A sequence[1] $s = (X_1 \dots X_l)$ is

[1]Each sequence in sequential patterns is considered as a rule.

TABLE I
PHARMACY DATABASE

Patient ID	Prescription Time	Medicine ID
1	July 20, 2005	S9255
1	July 25, 2005	S7230
2	July 9, 2005	S3925, S8756
2	July 14, 2005	S9255
2	July 20, 2005	S3925, S8256, S7230
4	July 25, 2005	S8756, S8256
4	July 29, 2005	S9255

TABLE II
DATASET REPRESENTATION

PID	TID	S3925	S7230	S8256	S8756	S9255
1	1	0	0	0	0	1
1	2	0	1	0	0	0
2	3	1	0	0	1	0
2	4	0	0	0	0	1
2	5	1	1	1	0	0
3	6	0	0	1	1	0
3	7	0	0	0	0	1

an ordered list of itemsets, and an itemset $X_i(1 \leq i \leq l)$ in a sequence is called a transaction. In a set of sequences, a sequence s is maximal if s is not contained in other sequences [12].

III. GENETIC ALGORITM

Genetic Algorithm (GA) is general purpose search algorithm which use principles inspired by natural genetic populations to evolve solutions to problems [15].

All GAs typically starts from a set, called population, of random solutions (candidate). These solutions are evolved by the repeated selection and variations of more fit solutions, following the principle of "survival of the fittest". The elements of the population are called individuals or chromosomes, which represent candidate solutions. Chromosomes are typically selected according to the quality of solutions they represent. To measure the quality of a solution, fitness function is assigned to each chromosome in the population. Hence, the better the fitness of a chromosome, the more possibility the chromosome has of being selected for reproduction and the more parts of its genetic material will be passed on to the next generations.

IV. MINING PHARMACY DATABASE SEQUENTIAL PATTERNS USING EVOLUTIONARY ALGORITHM

This section describes Genetic Algorithm, which is an Evolutionary Algorithm, for mining Sequential Patterns in Pharmacy Database. First, explanation of how the proposed algorithm represents the dataset, chromosome structure, and encoding scheme are presented. After that, used genetic operators are described. Then fitness assignment and selection criteria are defined. Finally, the algorithmic structure of the proposed algorithm is listed.

A. Dataset Representation

Datasets contain data which the algorithm is applied to, in order to discover Sequential Patterns. To allow an efficient counting, a representation method must be implemented. In the proposed algorithm, vertical bitmap representation is used to represent the dataset. Vertical Bitmap Representation (VBR) can be defined as follow.

Definition 2 [16]: VBR efficiently stores the transactional database as a series of vertical bitmaps. A vertical bitmap is created for each item in the dataset, and each bitmap has a bit corresponding to each transaction in the dataset. If item i appears in transaction j, then the bit corresponding to transaction j of the bitmap for item i is set to one; otherwise, the bit is set to zero. To enable efficient counting and candidate generation, divide the bitmap such that all of the transactions of each sequence in the database will appear together in the bitmap.

For example, using the pharmacy database in Table1, the Vertical Bitmap Representation can be established as in Table2, where PID is patient ID and TID is the transaction ID.

In VBR, if transaction T1 is before transaction T2 in a sequence, the index of the bit corresponding to T1 is smaller than the bit corresponding to T2. This propriety is important to know the order of the transactions without searching it.

B. Chromosome

This section discusses the used structure of GA chromosomes and how it is represented in this work.

1) Structure

In Pharmacy Database, every prescription, given a medicine-id value, has been recorded to the database. These medicine-id values are used for creating the chromosomes. In the proposed algorithm a fixed length chromosome is used, and its length is equal to number of medicines prescribed to patients. If Pharmacy Database is searched, the chromosome's shape should be as in Fig.1.

2) Representation

In Genetic Algorithm there are many alternatives to represent a chromosome based on other problem domains as in [6]–[8]. To decide which representation is better to be used for Sequential Pattern rules, David Goldberg [4] offered his advice saying *"The user should select a coding so that short, low-order schemata are relevant to the underlying problem and relatively unrelated to schemata over other fixed positions"* and *"The user should select the smallest alphabet that permits a natural expression of the problem"* [4]. The primary meaning behind these statements is that the proper choice of genetic representation is problem-dependent. If the application has a natural binary representation then binary is the best representation and if the application consists of integer variables then an integer representation may be appropriate.

In the proposed algorithm, binary is the most suitable representation because, comparing to the integer representation

Fig. 1. Chromosome structure.

S3925	S7230	S8256	S8756	S9255
1	0	1	1	1

Fig. 2. Chromosome representation.

which lost its advantage of canceling the encoding phase, it needs less space and it represents the needed information (element occurred or not).

For example, using the Pharmacy Database in Table.1, if a sequence is equal to <(S3925 S8256) (S9255) (S9255 S3925 S8756)>, it can be represented as in Fig.2.

Additionally, because of the unusual structure of Sequential Pattern rules, chromosomes representation became an issue. It is important in Sequential Pattern to encode the sequence of transactions. But, as it can be seen in Fig.2, it is impossible to extract the sequence order directly. To solve this problem it has been decided to associate the transactions sequences as metadata with each chromosome and consider it as a rule that represents the associate chromosome. This solution will take less time and space to complete the algorithm as a whole.

For example, using the pharmacy database in Table1, the metadata of sequence <(S3925 S8256) (S9255) (S9255 S3925 S8756)> that is represented in Fig.2 can be indexed as in Fig.3.

C. Genetic Operators

GA uses genetic operators to generate the offspring of the existing population. This section describes three operators that have been used in the proposed algorithm: selection, crossover, and mutation.

1) Selection

The selection operator chooses a chromosome in the current population according to fitness function and copies it, without changes, into the new population. The proposed algorithm uses Elitist selection, where the fittest members of each generation are copied into the next generation.

2) Crossover

The crossover operator, according to a certain probability, produces two new chromosomes from two selected chromosomes by swapping segments of genes. The proposed algorithm uses single-point crossover operation with

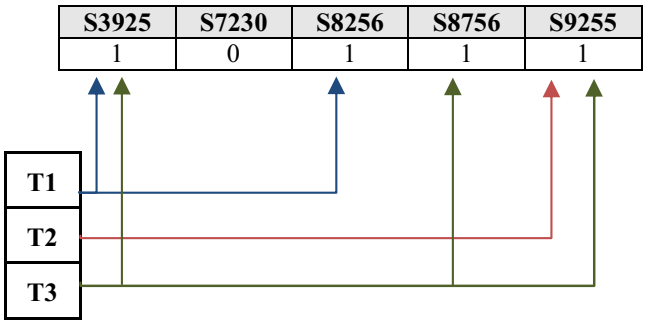

Fig. 3. Metadata representation.

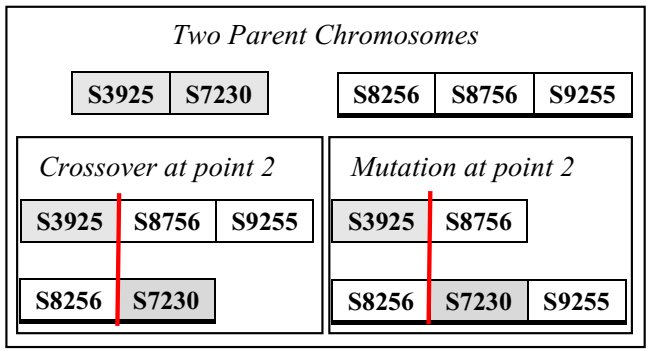

Fig. 4. Schematic representation of crossover and mutation operators.

probability of 0.7; chromosomes can be created as in Fig.4 [17].

3) Mutation

The mutation operator is used to maintain diversity. During the mutation phase, according to mutation probability, value of each gene in each selected chromosome is changed, as in Fig.4 [18]. The proposed algorithm use 0.001 as the mutation probability.

D. Fitness Function

The relationships in Sequential Patterns are resulting from applying some measures to determine and generate rules, called fitness function. There are a lot of evaluation rule measures in [9]–[11] that come from statistics, machine learning and data mining, each of them trying to evaluate one feature of the rule (precision, interest, reliability, comprehension, simplicity, etc.). Interestingness measures play an important role in data mining regardless of the kind of patterns being mined.

Piatesky-Shapiro, 1991 [6], proposed three principles that obeyed by any objective measure. Lenca et al. [6], proposed five properties, based on Piatesky principles, to evaluate the measures. Using these principles and property, "Sequential interestingness" measure is the most appropriate measure to be used as fitness function in the proposed algorithm since it applies most of the requirement; it can be defined as follows.

Definition 3 [10]: Sequential interestingness of sequence s is presented in Equation.1, where $(\alpha \geq 0)$ is a parameter defined by the user that represents how important the frequency of the pattern s and s_p is the transactions of sequence s.

$$\text{inst}(s) = \min_{s_p \in s}\{(Conf(s_p|s))^{\alpha}\} \times Supp(s) \qquad (1)$$

The first term of the equation evaluates that the frequencies of the transactions are not frequent while the second term evaluates that the frequency of the pattern is frequent[2]. In addition, parameter α is called the confidence priority, and the pattern that is bigger than or equal to the minimum sequential interestingness given by the user is called the interesting pattern.

[2]As well known, $Conf(s_p|s)$ divide number of s and s_p occurrence, in the database, over number of s occurrence; while $Supp(s)$ divide number of s occurrence, in the database, over number of all sequences available in the database [1], [2].

Input:
population size: N
Maximum number of generations: G
Confidence priority: α

Output:
Interesting sequence pattern: S

Algorithm:
1. Initialize counter $t = 0$
2. Generate population P_{old} of size N
3. For each chromosome $i \in P_{old}$
   3.1. Generate rule i from chromosome i
   3.2. Calculate fitness of rule i

4. Initialize empty population P_{new} of size N
5. For each chromosome i and $i+1 \in P_{old}$
   5.1. Mutate and crossover chromosome i with $i+1$
   5.2. Place reproduced chromosome in P_{new}
   5.3. Generate rule i from the reproduced chromosome
   5.4. Calculate fitness of rule i

6. Select fittest rules from P_{old}&P_{new} then place it in P_{old}
7. Set $t = t + 1$

8. If $(t > G)$ then
   $S = P_{old}$
   Stop algorithm
Else
   Go to Step 5

Fig. 5. Pseudo code of the proposed GA.

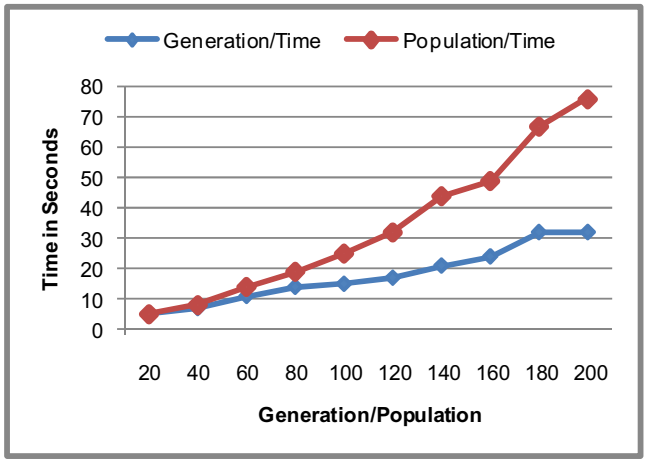

Fig. 6. Operation time related to generation and population size.

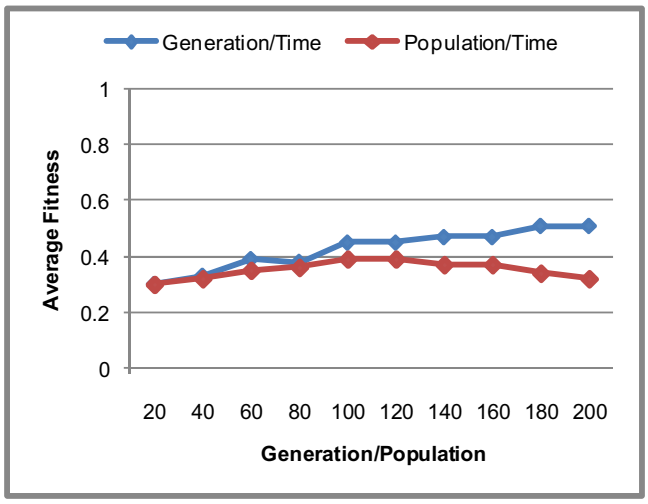

Fig. 7. Operation average fitness related to generation and population size.

E. GA for Mining Sequential Patterns

This section presents the algorithm that has been produced, as in Fig.5. After the encoding of the dataset, using "Vector Bitmap Representation" described in section IV.A, the population size, maximum number of generations, and confidence priority are taken as input; then the algorithm works as follow.

1) Loop counter is set to zero.
2) Initial population P_{old} is generated.
3) Every valid chromosome (with active elements, i.e. not only zeros) is associated with a sequential pattern rule, as described in section IV.B.2. Then, it calculates the rule fitness (using (1)).
4) Second population P_{new} is initialized with zeros; it will contain $P_{old's}$ chromosomes after applying GA operators, i.e. its children.
5) Population P_{old} is mutated and crossed over and then the resulting children is put into P_{new}, to associate the appropriate sequential pattern rule, as described in section IV.B.2, and calculate its fitness using (1). This step preserves the children of the population P_{old} in order to be compared using the selection operation in the next step.
6) Selection operation is conducted, between P_{old} and P_{new}, and the best rules chosen between the parents in P_{old} and the children in P_{new} are placed into P_{old}.
7) Increase the loop counter.
8) Check the termination condition. If the maximum generation number is reached, then stop the algorithm and put the P_{old} into S, which have the fittest rules conducted from the algorithm; otherwise, go to step 5.

V. Experiment Results

This section reports the results of the experiments conducted to analyze a Pharmacy Database. All the experiments were performed on 3 GHz Intel® Pentium®4 PC machine with 1.50 GB RAM, running Microsoft Windows XP. The algorithm is written with java in Borland JBuilder environment. During all of the tests, confidence priority, population, and generations are given by the user.

As experimental data, a real Pharmacy Database is taken from King Faisal Specialist Hospital and Research Centre, in KSA. The experiment was done on one year of heart

patients' prescriptions with 1361 transaction and 50 patients[3]. The crossover probability used is 0.7, while the mutation's is 0.001. The output of this experiment is a file that includes the interesting rules that represent the most suitable prescriptions sequences. For example, taken from the output file, a sample of the rules is as follows.

$$R1: \quad [S7230] \rightarrow [S6825, S8756] \qquad (F = 0.66)$$

$$R2: \quad [S7230][S7230] \rightarrow [S8756] \qquad (F = 0.70)$$

$R1$ tells us that when 100MG ASPIRIN is prescribed for patients, (%67) of these patients will take 5MG RENITEC with 50MG NORMOTEN afterwards, while $R2$ tells us that when 100MG ASPIRIN are prescribed two times in a row for patients, (%70) of these patients will take only 50MG NORMOTEN afterwards.

In the experiment, there are three parameters that must be determined: number of generation, population size, and confidence priority. Two experiments have been conducted in order to test the speed and fitness of the algorithm. First experiment set the population to 20 and the confidence priority to 0.5 with generation of [20...200]. The second, set the generation to 20 and the confidence priority to 0.5 with population of [20...200].

Comparing the two experiments, Fig.6 shows the time, in seconds, spent by the algorithm while Fig.7 shows the average fitness of the final output. Both Figures show the result related to increasing the generations and population size.

All these tests showed that when the generation or population increases the time will naturally increase. However, comparing the two experiments, as shown in Fig.6, GA takes less time when increasing the generation than increasing the population. Moreover, these tests also showed that when the generation increases, the average fitness will also increase. However, increasing the population does not guarantee increasing of the average fitness. Furthermore, average fitness of increasing the generation is much better than increasing of population.

From the experiment, it is observed that increasing of generation will take less time than increasing of population size. But, either ways, GA doesn't take along time; it is only a matter of seconds. In addition, increasing the population will not guarantee improvement of average fitness, some times it even make it worse, while increasing the generation most likely will give better average fitness.

At the end, from the results given above, it can be seen that even if GA reduces the time, it is important to choose the right generation and population size. Moreover, large population doesn't guarantee better performance, which means that one should be careful in choosing the right population size. In our opinion it is better to make the population size moderate while increasing the number of generations. This tradeoff will prospectively guarantee good rules in short time.

[3]It is better to study each field prescriptions separately, e.g., as done in the experiment, the study was applied on heart disease prescriptions; this is better than applying it on heart & cancer disease prescriptions together. This tradeoff will guarantee better results.

VI. Conclusion

In this paper Genetic Algorithm have been applied to find frequent sequences in Pharmacy Database in order to assess patient prescriptions and predict unusual activities to reduce prescription errors in timely manner. The algorithm utilizes the property of evolutionary algorithm that discovers new rules in a short time to overcome the shortage of pharmacists and contain the cost of healthcare delivery. The use of mutation in Genetic Algorithm makes the method capable of identifying global best even in very difficult problem domains. The method does not require knowledge about the distribution of the data, this way GAs can efficiently explore the space of possible solutions. The proposed algorithmic components including binary representation, selection, crossover and mutation operators all contribute to this excellent run time. Experimental results demonstrated that the proposed algorithm takes less time, and better fitness, when increasing the generation than increasing the population.

In the future, this algorithm will be tested on categorical Pharmacy Database in order to take the time gap and patients state into consideration, like the gender, age, disease, date and time, etc This way it will be possible to relate the prescriptions to the disease of the patients and the time between these prescriptions.

Acknowledgement

The authors acknowledge the cooperation of King Faisal Specialist Hospital and Research Centre, in KSA, for providing a real Pharmacy Database for this work.

References

[1] R. Agrawal and R. Srikant, "Mining sequential patterns," in *IBM Almaden Research Center*, 650 Harry Road, San Jose, CA 95120-6099, 1995.

[2] ——, "Mining sequential patterns: Generalizations and performance improvements," in *IBM Almaden Research Center*, 650 Harry Road, San Jose, CA 95120, 1996.

[3] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *American Association for Artificial Intelligence: AI Magazine*, pp. 37–54, 1996.

[4] D. Goldberg, *Genetic Algorithms*. Addison Wesley, 1989.

[5] F. Herrera, M. Lozano, and J. L. Verdegay, "Tackling real–coded genetic algorithms: Operators and tools for the behaviour analysis," *Artificial Intelligence Review*, vol. 12, pp. 256–319, 1998.

[6] J. Tay and D. Wibowo, "An effective chromosome representation for evolving flexible job shop schedules," in *The Genetic and Evolutionary Computation Conference (2)*, 2004, pp. 210–221.

[7] S. Wannarumon, "Aesthetic creation of endless forms: An application in jewelry design," pp. 395–410.

[8] Y. Zhou, *Study on Genetic Algorithm Improvement and Application*. Master thesis, May 2006.

[9] L. Geng and H. J. Hamilton, "Interestingness measures for data mining: A survey," *ACM Computing Surveys (CSUR)*, vol. 38, 2006.

[10] S. Sakurai, Y. Kitahara, and R. Orihara, "A sequential pattern mining method based on sequential interestingness," *International Journal of Computational Intelligence*, pp. 252–260, 2008.

[11] Q. Zhao and S. S. Bhowmick, "Sequential pattern mining: A survey," Nanyang Technological University, Tech. Rep. 2003118, 2003.

[12] M. Kaya and R. Alhajj, "Multi-objective genetic algorithm based approach for optimizing fuzzy sequential patterns," in *16th IEEE International Conference on Tools with Artificial Intelligence*, 2004.

[13] M. Leonard, M. Cimino, S. S. S. McDougal, J. Pilliod, and L. Brodsky, "Risk reduction for adverse drug events through sequential implementation of patient safety initiatives in a children's hospital," in *American Academy of Pediatrics*, vol. 18, no. 4, 2006, pp. 1124–1129.

[14] C. Antunes and A. L. Oliveira, "Sequential pattern mining algorithms: Trade-offs between speed and memory," in *Instituto Superior Tcnico/INESC-ID*, 2004.

[15] J. Wook and S. Woo, "New encoding/converting methods of binary GA/real-coded GA," *IEICE Transaction*, vol. E88-A, no. 6, pp. 1545–1564, 2005.

[16] J. Ayres, J. Gehrke, T. Yiu, and J. Flannick, *Sequential Pattern Mining using a Bitmap Representation*, 2nd ed. Alberta, Canada: SIGKDD, 2002.

[17] W. Spears and V. Anand, "A study of crossover operators in genetic programmin," in *Proceedings of the 6th International Symposium on Methodologies for Intelligent Systems*, 1991.

[18] W. Spears, "Crossover or mutation?" *Navy Center for Applied Research in Artificial Intelligence*, 1992.

[19] A. Freitas, *Computing Laboratory*. UK: University of Kent, 2008, ch. A Review of Evolutionary Algorithms for Data Mining.

[20] ——, *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Berlin: Spinger-Verlag, 2002.

[21] Y. Hirate and H. Yamana, "Generalized sequential pattern mining with item intervals," *Journal of computers*, vol. 1, no. 3, pp. 51–60, 2006.

[22] D. Olson and D. Delen, *Advanced Data Mining Techniques*. Berlin Heidelberg: Springer-Verlag, 2008.

[23] M. Pakhira and R. De, "Generational pipelined genetic algorithm (PLGA) using stochastic selection," *International Journal of Computer Systems Science and Engineering*, vol. 4, no. 1, pp. 75–88, 2007.

[24] C. Romero, S. Ventura, and P. Debra, *Knowledge Discovery with Genetic Programming for Providing Feedback to Courseware Authors*. Netherlands: Kluwer Academic Publishers, 2004.

[25] D. Taniar, *Data Mining and Knowledge Discovery Technologies*. New York: Hershey, 2008.

[26] S. Y. W. Li, "Paper survey on sequential pattern data mining," December 2004.