Journal of Universal Computer Science, vol. 15, no. 4 (2009), 742-764
submitted: 30/6/08, accepted: 27/2/09, appeared: 28/2/09 © J.UCS

Integrative Discovery of Multifaceted Sequence Patterns by Frame-Relayed Search and Hybrid PSO-ANN

Sing-Wu Liou
(National Yunlin University of Science and Technology, Yunlin, Taiwan
Graduate School of Engineering Science and Technology
g9110808@yuntech.edu.tw)

Chia-Ming Wang
(National Yunlin University of Science and Technology, Yunlin, Taiwan
Graduate School of Engineering Science and Technology
g9410805@yuntech.edu.tw)

Yin-Fu Huang[1]
(National Yunlin University of Science and Technology, Yunlin, Taiwan
Graduate School of Computer Science and Information Engineering
huangyf@yuntech.edu.tw)

Abstract: For de novo pattern mining in genomic sequences, the main issues are constructing pattern definition model (PDM) and mining sequence patterns (MSP). The representations of PDMs and the discovery of patterns are functionally dependent; the performances thus depend on the adopted PDMs. The popular PDMs provide only descriptive patterns; they lack multifaceted considerations. Many of existing MSP methods are tied up with the exclusively devised PDMs, and the specialized and sophisticated models make the mined results hard to be reused. In this research, an integrative pattern mining system is proposed, which consists of a computation-oriented PDM (CO-PDM) and general-purpose MSP (GP-MSP) methods. The CO-PDM defines four computational concerns (CCs) as facets of MSP: expression (E), location (L), range (R) and weight (W), which are integrated into a frame-relayed pattern model (FRPM). The GP-MSP develops a frame-relayed search strategy to resolve the ELR-CCs firstly, with the aids of critical-parameter automating (CPA) procedure; and then the W-CC is determined by hybridizing particle swarm optimization (PSO) and artificial neural network (ANN). The proposed FRPM and GP-MSP had been implemented and applied to 22,448 human introns; from the results, all the well-known patterns were recovered and some new ones were also discovered. Furthermore, the effectiveness of identified patterns were verified by a two-layered k-nearest neighbor (k-NN) classifier; the average precision and recall are 0.88 and 0.92, respectively. By the case study, the integrative PDM-MSP system is believed to be effective and reliable; it is optimistic the proposed CO-PDM and GP-MSP are both widely applicable and reusable for mining sequence patterns in the eukaryotic protein-coding genes.

Key words: pattern mining, multifaceted sequence patterns, computation-oriented pattern definition model, computational concerns, frame-relayed pattern model.

Category: J.3, I.2.4, I.2.6, I.5.2

[1] Corresponding author

1 Introduction

For *de novo* pattern mining in gene sequences, it is usually divided into two sub-problems: site representation and site discovery [Stormo, 2000], or similarly, constructing pattern models and devising mining algorithms [MacIsaac and Fraenkel, 2006]. In this research, the two sub-problems are defined as constructing pattern definition model (PDM) and mining sequence patterns (MSP), respectively. In terms of PDMs, they were devoted to constructing either *qualitative* or *quantitative* model for describing patterns; while in terms of MSP, the major tasks are devising pattern-search methodologies and defining the thresholds of being significant according to the adopted or specially devised PDMs. The limitations on existing pattern mining algorithms had been revealed [Hu et al., 2005]; thus substantial improvements in both PDM and MSP are in demand.

The *qualitative*-PDMs, such as consensus [Pribnow, 1975], position weight matrix (PWM) [Stormo et al., 1982] and regular expression [Brazma et al., 1998], provide *descriptive* patterns; the *quantitative*-PDMs, such as entropy [Schneider et al., 1986] and information content [Berg and von Hippel, 1987], reduce the sequence context to numerical results. The PWM had been modified [Sinha, 2006] and and information content had been refined [Reddy et al., 2006] in search of better performance; some specially devised PDMs, such as subtle motif model [Keich and Pevzner, 2002] and maximum likelihood framework [Chen et al., 2007], had also been proposed. However, all the above-mentioned PDMs are *perception-oriented*, that means the important pattern-related information, such as possible lengths and positions of patterns or the formats for expressing patterns, was independently considered. The applicability of existing PDMs are complementary to each other; yet, their formats are not compatible and the contents are not joinable. Aiming at widely applicable for *in silico* analysis, the *computational concerns* (CCs) are proposed to integrate the advantages of existing PDMs.

Consensus and PWM are the most popular PDMs; they are widely used for pattern-finding [Day and McMorris, 1992] and for predicting the transcription factor binding sites [Chen et al., 2007, Sinha, 2006]. Four multifaceted CCs are derived from them; three CCs are firstly identified: *expression* (*E*-CC), *location* (*L*-CC) and *range* (*R*-CC). *E*-CC is comprehensive descriptions, *L*-CC is the relative positions in sequences and *R*-CC is the extent of spreading. It seems the *ELR*-CCs are intuitive and can be easily derived; however, there are two new ideas behind them. Firstly, they are integrated to construct a *computation-oriented* PDM (*CO*-PDM); secondly, they are *pre-defined* concerns that make them intrinsically different from the *post-derived* attributes in classical PDMs.

The basis for identifying *ELR*-CCs is the over-threshold statistical significance; accordingly, two issues should be addressed in advance to make the mining results more useful and more practical. One is determining the threshold of being significant and the other is harmonizing the statistical significance with biological significance. The statistical significances are usually used as biological significance implicitly in computational analyses; however, there are gaps between them when the effectiveness for

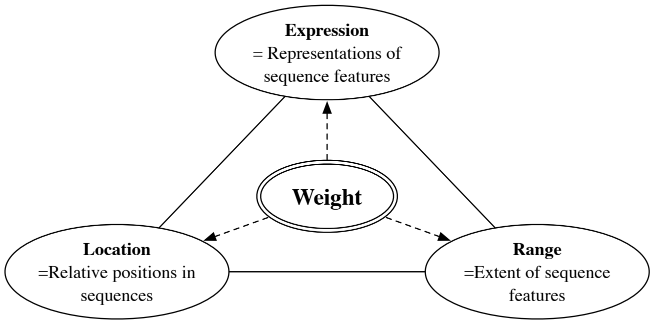

Figure 1: *CO-PDM: computation-oriented pattern definition model.*

real-world applications is considered [Altschul et al., 1994]. Statistical significances are *pre-judged* weights without testing the effectiveness and this may lead to take risks in biased decisions. Whether the mining results are also biologically significant or not should be verified by real-world data to approve their effectiveness. Hence, for making sure the identified multifaceted sequence patterns are also biologically meaningful, the weight (*W*-CC) is proposed. In this work, the *W*-CC is derived independently after identifying the *ELR*-CCs, thus the *W*-CC is *post-judged* weight. All the *WELR*-CCs are integrated into a four-faceted *CO*-PDM as briefly depicted in Figure 1.

For resolving the WELR-CCs, a frame-relayed pattern model (FRPM) was constructed. Based on the FRPM, the *ELR*-CCs are identified with frame-relayed search (FRS); and the *W*-CC is determined by sensitivity analysis realized by hybridizing particle swarm optimization (PSO) and artificial neural networks (ANN). The potential sequence patterns are divided into two categories, the uni-frame patterns (UFPs) and the multi-frame patterns (MFPs), based on their frequencies and distributions in sequences. The significant UFPs (SUFPs) focus on *vertical* distribution of tandem repeats and significant MFPs (SMFPs) focus on the horizontal ones as illustrated in Figure 2.

The MSP methods are PDM-dependent; therefore, their algorithms vary with the adopted or specially devised PDMs. Defining the thresholds of being significant is the common core of MSP algorithms; yet, this task is usually solved by critical-parameter templates (CPTs). The users have to specify some result-sensitive parameters, such as minimum pattern length and minimum frequency of appearance in sequences, listed in CPTs; thus, the mining results vary with specified critical parameters. For example, three CPT-based pattern-finding programs, Consensus [Pribnow, 1975], AlignACE [Roth et al., 1998] and Bioprospector [Liu et al., 2001], can be found in the tool suite BEST [Che et al., 2005]. To improve the unpredictable variance in results by using CPTs, the critical-parameter automating (CPA) mechanism was developed.

For integrating PDM and MSP, available MSPs were tied up with their exclusive PDMs; for examples, the covariance model in CMfinder [Yao et al., 2006], the maxi-

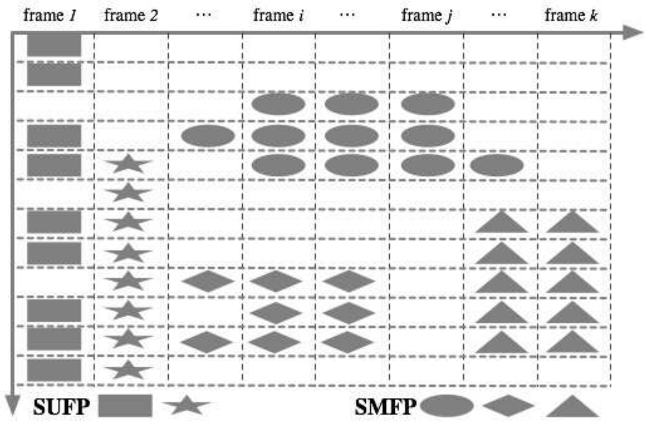

Figure 2: *Tandem repeats of codons form the SUFPs and SMFPs.*

mum density subgraph in MotifCut [Fratkin et al., 2006] and Markov background models in Bioprospector [Liu et al., 2001]. The exclusive PDMs embedded in MSPs make both of them lack the capability of being applied independently; moreover, the specialized and sophisticated PDMs are hard to reuse. Therefore, both the applicability and reusability are considered in designing the mining system for integrative discovery of multifaceted sequence patterns.

Finally, the proposed four-faceted PDM and CPA-equipped MSP are integrated into a general-purpose pattern-mining system; for validation, a case study on 22,448 human introns was performed after implementation. From the results, all the well-known sequence patterns were recovered and some new patterns were also revealed; therefore, both the reliability and effectiveness of the patterns with *ELR*-CCs were verified. Furthermore, the significance of specially derived *W*-CC was tested by a two-layered *k*-nearest neighbor (*k*-NN) classifier; for patterns with *W*-CC, the average *precision* and *recall* was 88% and 92%, respectively, which outperform the unweighted ones; thereby, the high performance in a simple classifier demonstrated the effectiveness of *W*-CC. It is believed the proposed four-faceted *CO*-PDM (i.e. *WELR*-CCs) and the *general-purpose* MSP are useful tools for mining sequence patterns; it is also promised both of them are widely applicable in the eukaryotic protein-coding genes.

In this work, a methodology combining frame-relayed search and hybrid PSO-ANN is proposed for discovering the multifaceted sequence patterns. The details of the devised methodology are described in Section 2; the experimental results by applying the methodology to real data are presented in Sections 3. Some viewpoints about the proposed methodology and the experimental results are discussed in Section 4 and conclusions are given in Section 5.

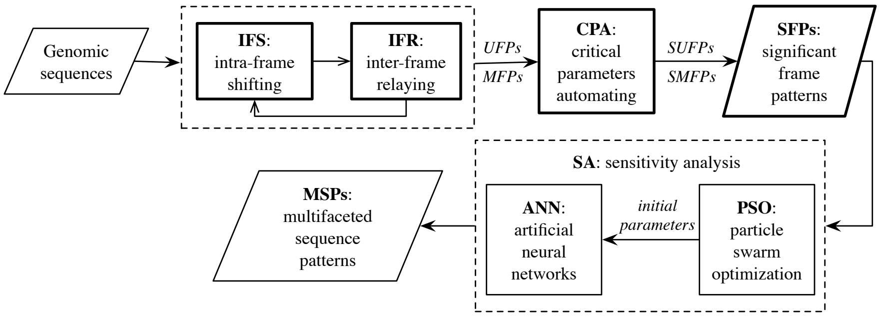

Figure 3: *Procedures for discovering multifaceted sequence patterns.*

2 Method

The multifaceted sequence patterns are determined by two consecutive processes: discovery of significant frame-patterns (SFPs) and deciding weights of SFPs; the complete mining processes are shown in Figure 3. The *intra*-frame shifting (IFS) and *inter*-frame relaying (IFR) are responsible for resolving the *ELR*-CCs, and the thresholds of being statistical significance for each of *ELR*-CCs are defined by the critical parameters automating (CPA) processes; while the *W*-CC is decided by sensitivity analysis (SA) realized by a PSO-initialized ANN. The final results are defined as multifaceted sequence patterns, which provide four computational concerns: *weight, expression*, *location* and *range*.

2.1 Framing the sequence

By adopting the concept of *frameshift* from the researches in *reprogramming of mRNA translation*, each sequence is partitioned into frames. The single *frame* has to accommodate a codon (i.e., 3 bps) and to provide the space for ± 1 *frameshift* (i.e., 2 extra bps); accordingly, the size of a frame is defined to be 5 bps as illustrated in Figure 4. Codons are often used to represent the sequence composition of coding sequences; according to the codon-triplet context, we extend the meaning of *codon* to *tri-nucleotides* in this research. Therefore, for the proposed pattern-mining methodology, a *codon* stands for *tri-nucleotides* in the sequence. Thus, the proposed methods are widely applicable; they can be applied to all kinds of nucleotide sequences including the coding and non-coding sequences.

2.2 Frame-relayed pattern model (FRPM)

For pattern discovery in sequences, a frame-relayed pattern model (FRPM) shown in Figure 5 was constructed to retrieve *ELR*-CCs related information. FRPM defines the

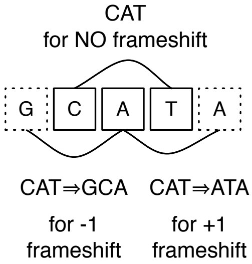

Figure 4: *The 5-bps frame: codon and* ± 1 *frameshift.*

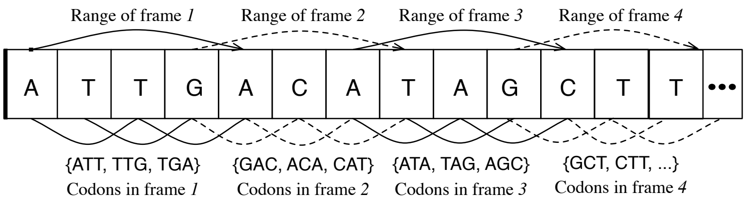

Figure 5: *FRPM: Frame-relayed pattern model.*

unit of each computational concern: the unit of *E*-CC is codon; the unit of *L*-CC is the start position of frames; and the unit of *R*-CC is a 5-bps frame. From the illustrative example in Figure 5, it can be seen there are three codons within each frame; and the last and first 2 bps are overlapped in contiguous frames.

The first distinguishing property of FRPM is the *frameshifting* analyses which keeps both the specificity of intra-codon nucleotides and the inter-codon repeats; taking the sequence GTAAGCCCTTACAG as an example, it can be observed, from the Figure 6, that the sets of codon-triplets are different in the frameshifted sequences. For keeping all the inter-codon information, two specially devised counting techniques, the *in-frame codon counting* and *codon stepping*, are developed to preserve the detailed and specific information about all the codon composition in each frameshifting for resolving the *ELR*-CCs; for clarity, a shorter example sequence ACGTACGT is provided to introduce the processes of intra-frame shifting (IFS) *by in-frame codon counting* and inter-frame relaying (IFR) by *codon stepping* as illustrated in Figure 7. IFS counts the frequencies of all the 64 standard codons within a specific frame for all the input sequences. For the example sequence ACGTACGT, the three codons in the first frame are $\{ACG_0, CGT_1, GTA_2\}$, where the subscriptions are *in-frame* positions as shown in Figure 7. After IFS, the *codon-stepping* performs a 3-bps jump to move to the next frame and *relays* the IFS process.

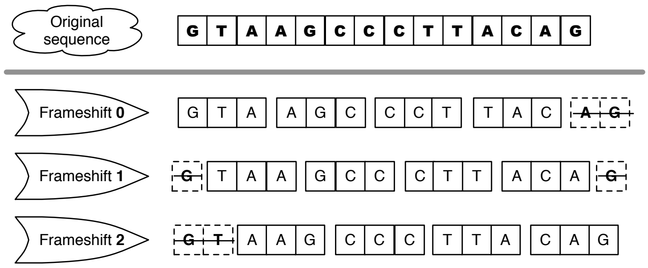

Figure 6: Different codon-triplets in frameshifted sequences.

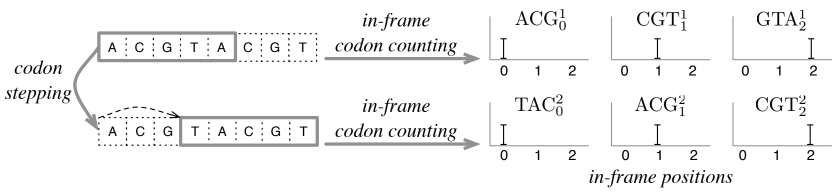

Figure 7: *In-frame codon counting and codon stepping.*

The information about the distribution of codons is recorded in the codon distribution matrix (CDM) shown in Figure 8; the value c_j^i in CDM is the ID of the codon (CID, ranging from 1 to 64) which appears within frame i at *in-frame* position j. The corresponding CDM for the sequence in Figure 7 is shown in Table 1. In addition, the codon-repeats matrix (CRM) shown in Figure 9 is devised to record the tandem repeats of one specific codon; an example sequence set {GTAGGAGT, GTAGGAAG, GTAGGCAG} is used to demonstrate the usage of CRM as shown in Table 2. Both the CDM and CRM are important infrastructures in mining multifaceted sequence patterns. The

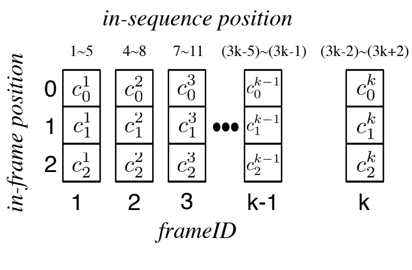

Figure 8: *CDM: codon distribution matrix.*

in-sequence positions	1-5	4-8
in-frame position *0*	7(ACG)	50(TAC)
in-frame position 1	28(CGT)	7(ACG)
in-frame position *2*	45(GTA)	28(CGT)
frameID	*1*	*2*

Table 1: *CDM of example sequence ACGTACGT.*

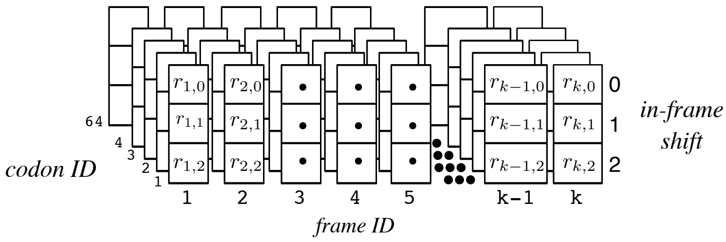

Figure 9: *CRM: codon-repeats matrix.*

CDMs and CRMs derived from the data set provide the computational basis for identifying significant UFPs and MFPs (i.e., the SUFPs and the SMFPs);

2.3 Uni-frame pattern (UFP) and Multi-frame pattern (MFP)

By iterating the IFS and IFR processes on all the sequences, 64 *codon pillars* (CPs) in each frame will be generated. Without loss of generality, a classical codon pillar (CP) for one specific codon is depicted in Figure 10, where c_x is the total codon count at in-frame position x. Then, the values $\{c_0, c_1, c_2\}$ are further divided into two parts: one is uni-frame pattern (UFP) and another is multi-frame pattern (MFP); UFP is defined as the vector $(c_0 - c_1,\, c_1 - c_2)$; and MFP is defined as the minimum CP, MIN(c_0, c_1, c_2). UFPs are used to identify *well*-aligned and *ill*-aligned codons highly expressed within *one* frame; and MFPs are used to identify the clusters of moderately expressed tandem repeats across *contiguous* frames.

2.4 Critical parameters automating (CPA)

The criteria for determining whether an UFP or a MFP is *significant* or not are the core of MSP methods; moreover, automatically determining the thresholds of being *significant* is a key to bridge PDMs and MSPs; thus, an idea of critical-parameter automating (CPA) was motivated. For making sure the final discovered UFPs and MFPs are truly meaningful, a heuristic *high-value* detection method is motivated by using $\mu + \sigma$ (μ

GTAGGAGT
GTAGGAAG
GTAGGCAG

GTA		
frame ID	1	2
$shift_0$	**3**	0
$shift_1$	0	0
$shift_2$	0	0

TAG		
frame ID	1	2
$shift_0$	0	0
$shift_1$	**3**	0
$shift_2$	0	0

AGG		
frame ID	1	2
$shift_0$	0	0
$shift_1$	0	0
$shift_2$	**3**	0

GGA		
frame ID	1	2
$shift_0$	0	2
$shift_1$	0	0
$shift_2$	0	0

GAG		
frame ID	1	2
$shift_0$	0	0
$shift_1$	0	1
$shift_2$	0	0

AGT		
frame ID	1	2
$shift_0$	0	0
$shift_1$	0	0
$shift_2$	0	1

AAG		
frame ID	1	2
$shift_0$	0	0
$shift_1$	0	0
$shift_2$	0	1

GGC		
frame ID	1	2
$shift_0$	0	1
$shift_1$	0	0
$shift_2$	0	0

GCA		
frame ID	1	2
$shift_0$	0	0
$shift_1$	0	1
$shift_2$	0	0

CAG		
frame ID	1	2
$shift_0$	0	0
$shift_1$	0	0
$shift_2$	0	1

GAA		
frame ID	1	2
$shift_0$	0	0
$shift_1$	0	1
$shift_2$	0	0

Table 2: *The CRMs in a set of sequences*

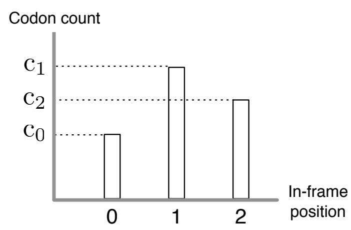

Figure 10: *CP: codon pillar of a codon.*

is the mean and σ is the standard deviation) as the outlier boundary; for a normal-distribution like data set, according to the 68-95-99.7 rule, $\mu+\sigma$ will include the top 32% of values (i.e., the *High* values). Basing on the $\mu+\sigma$ mechanism, an *iterative* $\mu+\sigma$ process was proposed to realize the CPA concept; it will highlight the *over-expressed* frame patterns (FPs, i.e., UFPs or MFPs). The CPA goes as follows: assuming μ and σ are the mean and standard deviation of FPs, for $x \in FP_s$, it is firstly classified as either Low ($x < \sqrt{2}*(\mu_1-\sigma_1)$), Medium ($\sqrt{2}*(\mu_1-\sigma_1) \leq x \leq \sqrt{2}*(\mu_1+\sigma_1)$) or High ($x > \sqrt{2}*(\mu_1+\sigma_1)$ as illustrated in Figure 11. For x classified as High, it is a significant FP (SFP), and therefore forms a SFP candidate set C^1_{SFP} for the 1^{st} round. Then, the demand for 2^{nd} or even the third round of CPA on C^1_{SFP} and C^2_{SFP} depends on applications.

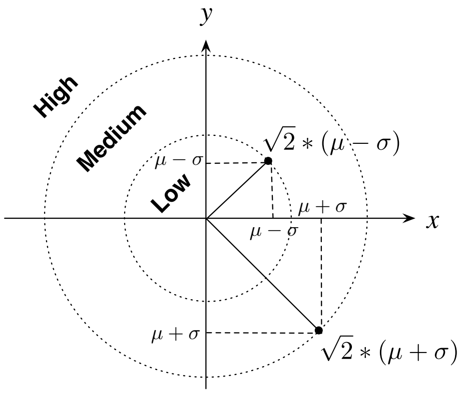

Figure 11: *CPA: critical parameters automating.*

2.5 The significant UFP (SUFP) and significant MFP (SMFP)

The significance of the UFP vectors are defined as the distance to the origin, i.e. the DIS_{ufp}; accordingly, $DIS_{ufp} = \sqrt{(c_0 - c_1)^2 + (c_1 - c_2)^2}$ (ref. to Figure 10). Similarly, DIS_{mfp} is defined as $\sqrt{(High\_Frames)^2 + (Cont\_Hframes)^2}$, where the $High\_Frames$ is the number of frames classified as High, and the $Cont\_Hframes$ is the maximum number of contiguous frames all classified as High. By applying the CPA process, it takes a single round to determine the significant UFPs (SUFPs) in DIS_{ufp}; while it takes two rounds to decide the final significant MFPs (SMFPs). The reason for single round only in identifying SUFPs is evident; yet in SMFP decision processes, the 1^{st} round CPA is used to determine the $High\_Frames$ and $Cont\_Hframes$; and the 2^{nd} round CPA decides the over-expressed DIS_{mfp} as significant MFPs (SMFPs). The special CPs for SUFPs and SMFPs are introduced in Figure 12; the characteristics of SUFPs are High-DIS_{ufp} and Low-DIS_{mfp}; on the contrary, High-DIS_{mfp} and Low-DIS_{umfp} reveal the SMFPs.

2.6 PSO-initialized ANN

Artificial neural networks (ANN) are robust and general methods for function approximation, prediction, and classification. The superior performance and generalization capabilities of NN have attracted much attention in the past thirty years. Backpropagation (BP) algorithm [Werbos, 1974](i.e., the most famous learning algorithm of NN) has been successful applied in many practical problems. The learning problem can be though as searching through a hypothesis for the one best fit the training instances [Mitchell, 1997]; however, the random initialization mechanism of NN might cause the

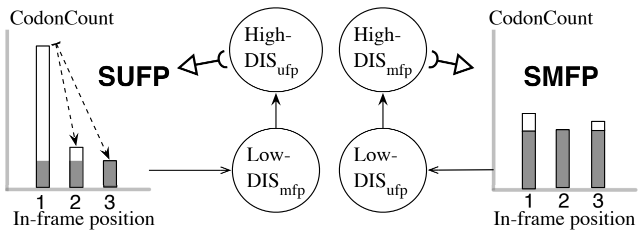

Figure 12: *Classic CPs for SUFPs and SMFPs.*

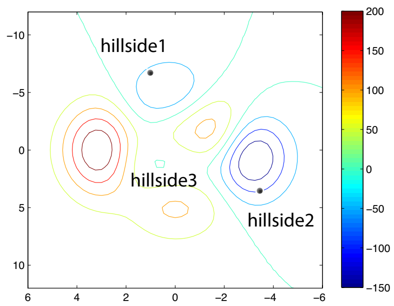

Figure 13: *Optimization surface.*

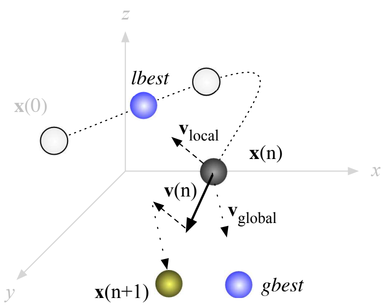

Figure 14: *The position update of PSO.*

optimum search process failed and return an unsatisfied solution, since BP is a local-search learning algorithm [Rumelhart et al., 1986]. For example, once the random initialization of the synaptic weights leads to the search process starting from hillside 1, as shown in Figure 13, BP algorithm would update the synaptic weights and go along the gradient direction. Consequently, it seems hopeless to reach a better solution near the global optimum in valley 2. Therefore, lots of trials and errors are a general guideline in most practical usages.

Nevertheless, the particle swarm optimization (PSO), a population-based evolutionary algorithm proposed by [Kennedy and Eberhart, 1995], has different characteristics. The search task is cooperated with a group of particles (solutions) in an n-dimensional space for an n-variable objective function via communicating with others. Information of the personal best position (local view) and the global best position among particles (global view) would be kept and used to update the next position. Therefore, PSO exhibits a nice global search property and seldom falls into a trap. As shown in Figure 14, the particle moves from position $\mathbf{x}(0)$ and halts at position $\mathbf{x}(n)$ in the hyperspace.

The new position of the $(n+1)^{th}$ iteration would be updated according to equation 1 [Eberhart and Shi, 1998]:

$$x(n+1) = x(n) + [\underbrace{mo \times \mathbf{v}(n)}_{present} + \underbrace{c_1 \times r \times \mathbf{v}_l}_{local\,view} + \underbrace{c_2 \times r \times \mathbf{v}_g}_{global\,view}] \tag{1}$$

where mo is the moment term which gradually reduces from 1 over iterations, $\mathbf{v}(n)$ and $\mathbf{x}(n)$ are the velocity and position in the n^{th} iteration respectively, c_1 and c_2 are positive constants, and r is a random number between 0 to 1. However, one might argue that how appropriate constants are decided and what the influence is. Although the parameters setting really has some influences on the precision of solutions, we adopt an experience value 2. Despite its lack of precision, PSO still provides acceptable solutions due to its parallel processing and global search characteristics. According to these observations, we are motivated to combine the advantages of NN and PSO together. PSO is used as an initializer of NN; i.e., the generator of initial synaptic weights of NN. In other words, the lowest valley in Figure 13 is first found by PSO, and then a gradient-descent based NN would go down carefully to obtain a precise solution. Finally, a sensitivity analysis is conducted on the well-trained network to estimate the relative importance of input attributes.

2.7 Sensitivity analysis

Sensitivity analysis (SA) is a common technique to realize the relationships between input and output variables; it could be used to check the quality of a hypothesis model as well. The idea behind SA is to slightly alter the input variables, and then the corresponding responses with respect to the original ones would reveal the significance of the variables. Therefore, the most important part of SA is to determine the adequate measurements as *disturbance* of input variables. Although applying SA to neural networks had been studies in some works [Yoon et al., 1994, Steiger and Sharda, 1996], their purposes were usually identifying important factors only; while we go one step further, in this work, not only significant input attributes will be recognized but also the relative importance of them will be estimated. We proposed a new measurement, *disturbance*, for the relative sensitivity. The elements of disturbance instances used in the SA are defined as follows:

$$x_m = \begin{cases} (1 \otimes d) \times x_m, & if\ m = j \\ x_m, & otherwise \end{cases}, \ \forall x_m \in \mathbf{x}^i_{j\uparrow}, \tag{2}$$

where $\mathbf{x}^i_{j\uparrow}$ is the i^{th} instance in the training set, with the j^{th} attribute *increased* according to the disturbance ratio d; i.e., the symbol $\otimes$ denotes a plus sign. In other words, except the j^{th} attribute, all other attributes of the i^{th} instance are fixed. Similarly, $\mathbf{x}^i_{j\downarrow}$ is with the j^{th} attribute *decreased*; i.e., the symbol $\otimes$ denotes a minus sign. The relative

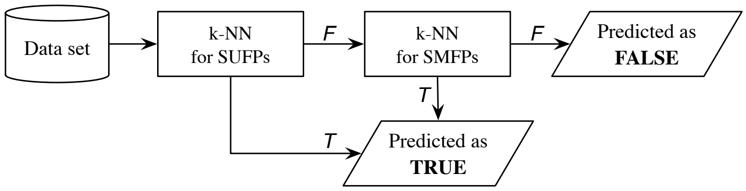

Figure 15: *Two-layered* k-*NN classifier.*

sensitivity of j^{th} attribute is defined as follows:

$$rs_j = \frac{\sum_{i=1}^{p}\left(\left|net(\mathbf{X}^i_{j\uparrow})-net(\mathbf{X}^i_j)\right|+\left|net(\mathbf{X}^i_{j\downarrow})-net(\mathbf{X}^i_j)\right|\right)}{\min_j\left\{\sum_{i=1}^{p}\left(\left|net(\mathbf{X}^i_{j\uparrow})-net(\mathbf{X}^i_j)\right|+\left|net(\mathbf{X}^i_{j\downarrow})-net(\mathbf{X}^i_j)\right|\right)\right\}} \tag{3}$$

where the function *net* is the trained network, and the relative sensitivity (rs) is normalized by the minimal sensitivity attribute among all attributes. Finally, the weights of patterns (i.e., the W-CC) are determined according to rs_j.

2.8 Two-layered *k*-NN classifier

In order to verify the effectiveness of discovered patterns, a two-layered *k*-nearest neighbor(*k*-NN) classifier [Cover and Hart, 1967] was constructed. The Euclidean distance in original *k*-NN was modified to a weighted Euclidean distance as Equation 3, where n is the number of dimensions, and $\mathbf{w}_i$, $\mathbf{x}_i$ and $\mathbf{x}'_i$ are the i^{th} attributes of weight vector $\mathbf{w}$, training instance $\mathbf{x}$ and testing instance $\mathbf{x}'$, respectively.

$$d(\mathbf{x},\mathbf{x}') = \sqrt{\sum_{i=1}^{n}\mathbf{w}_i(\mathbf{x}_i-\mathbf{x}'_i)^2} \tag{4}$$

The experiment was carried out with the 10-fold cross validation for each specific k (i.e., the k closest neighbor). First, the whole sequence was randomly divided into 10 divisions with the equal size. The class in each division was represented in nearly the same proportion as that in the whole data set. Then, each division was held out in turn and the remaining nine-tenths were directly fed into the two-layered *k*-NN classifier as the training instances. Since every sequence could be expressed as two parts: SUFPs and SMFPs, and for validating the effectiveness of *WELR*-CCs, a two-layered *k*-NN classifier was constructed. The first level *k*-NN classifier select candidates based on the SUFPs; subsequently, the final judgment was made by the second level *k*-NN based on the MFPs. The flow chart of two-layered *k*-NN classifier is shown in Figure 15.

3 Experimental results

Two criteria were used to choose the experimental sequences: existing both well-known patterns and *pattern-deserts* in sequences. To our knowledge, the introns in genes meet the prerequisite; four *cis*-acting splicing signals inside introns provide special sequence context for spliceosome to recognize the splice sites: 5' splicing site (5SS), 3' splicing site (3SS), branch point (BP) and poly-pyrimidine tract (PPT) [Sharp, 1987]; 5SSs and 3SSs define the exon-intron junctions; BPs initiate the lariat formation and PPTs facilitate the exon ligation. Their sequence patterns were all well described in the intron definition model (IDM) [Moore, 2000, Patel and Steitz, 2003]; as for the *pattern-deserts*, they can also be easily seen in the downstreams of 5SSs and upstreams of 3SSs from the classical IDM. Thus, the introns are selected as a case study for verifying the effectiveness and reliability of proposed methods in mining sequence patterns

3.1 Data set

By closely observing the ever proposed IDMs, it seems the intron flanks of splice site are much important than the exon flanks in terms of splicing because all the well-known sequence patterns are located inside introns; thus, the intronic flank sequences of splice sites (i.e. the introns) were chosen as the experimental sequences. For working on 5-bps frame and cooperating with the *codon-stepping* strategy, the length of intron sequences was set to 101 bps; and then, 101 bps downstream of the 5SS and 101 bps upstream of the 3SS were extracted separately. For complete analyses, all introns in human chromosome 1 (from NCBI human genome build 36.2) were extracted; both the 5SS and 3SS data set finally comprised 22,448 introns. The average length of human introns is 2364 bps [Gopalan et al., 2004], and in the manually collected human introns, only 4.5% of them are shorter than 101 bps. Without loss of the generality, the introns shorter than 101 bps were ignored in the experiment. For validating the statistical significance and biological significance, a false data set, the GT_RND_AG, was constructed. GT_RND_AG was generated by random sequences with GT-leadings for the 5SS sequences and AG-tailings for the 3SS sequences; the GT_AG_RND stands for the pseudo introns.

3.2 The SUFP and SMFP with WELR-CCs

All of the SUFPs and SMFPs discovered are listed in Table 3 and 4 respectively; and the information about the *WELR*-CCs was clearly specified. For highlighting the most valuable SUFPs and SMFPs, the CPA is applied to the *W*-CCs (obtained by Equation 3) and the weights worthy of special notice (i.e., the weights over the $\mu_{Wcc} + \sigma_{Wcc}$) are shown in boldface. The *W*-CCs of SUFPs and SMFPs were estimated independently; hence, their weights should be examined and used separately.

5SS					3SS				
ID	Weight	Expression	Location	Range	ID	Weight	Expression	Location	Range
1	**0.66**	AAG	1	5	7	**0.39**	ACA	1	5
2	0.18	GAG	1	5	8	**1.00**	CAG	1	5
3	**0.55**	GTA	1	5	9	0.05	GCA	1	5
4	0.23	TAA	1	5	10	**0.32**	TAG	1	5
5	0.19	TGA	1	5	11	0.24	TCA	1	5
6	**0.64**	AGT	4	5					

Table 3: *SUFPs of 5SS and 3SS.*

5SS					3SS				
ID	Weight	Expression	Location	Range	ID	Weight	Expression	Location	Range
1	0.01	CTG	7	5	20	0.13	TCT	4	5
2	0.08	GGG	7	5	21	**0.20**	TTT	4	5
3	0.05	TTT	7	5	22	0.08	ATT	7	17
4	0.11	CTG	10	17	23	0.10	CTG	7	17
5	**0.26**	GGG	10	17	24	**0.73**	TCT	7	17
6	0.08	TCT	10	17	25	**0.31**	TGT	7	17
7	0.05	TGG	10	17	26	**0.95**	TTT	7	17
8	**0.18**	TTT	10	17	27	**0.50**	AAA	22	47
9	**0.49**	AAA	25	68	28	**0.19**	ATT	22	47
10	**0.31**	CTG	25	68	29	**0.27**	CTG	22	47
11	**1.00**	GGG	25	68	30	**0.35**	TCT	22	47
12	**0.26**	TCT	25	68	31	**0.20**	TGT	22	47
13	**0.16**	TGG	25	68	32	**0.58**	TTT	22	47
14	**0.68**	TTT	25	68	33	0.05	AAA	67	8
15	0.07	AAA	91	11	34	0.05	CTG	67	8
16	0.07	CTG	91	11	35	0.04	TCT	67	8
17	0.11	GGG	91	11	36	0.05	TTT	67	8
18	0.04	TCT	91	11	37	0.02	AAA	73	5
19	**0.19**	TTT	91	11	38	0.01	ATT	73	5
					39	0.02	CTG	73	5
					40	**0.27**	AAA	76	26
					41	0.08	ATT	76	26
					42	**0.23**	TTT	76	26

Table 4: *SMFPs of 5SS and 3SS.*

3.3 Recovery of well-known patterns

According to the classical intron definition model [Moore, 2000], the consensus of 5SS and 3SS are 5'-$G_1T_2R_3A_4G_5T_6$-3' (R=[A/G]) and 5'-$N_4Y_3A_2G_1$-3' (Y=[C/T], N=[A/G/C/T]) respectively where the subscripts of consensuses are the relative positions to 5' and 3' ends of sequences. By decomposing 5SS and 3SS consensuses with codons, the composition is {GTA_{1-3}, GTG_{1-3}, TAA_{2-4}, TGA_{2-4}, AAG_{3-5}, GAG_{3-5}, AGT_{4-6}} and {GTA_{1-3}, GTG_{1-3}, TAA_{2-4}, TGA_{2-4}, AAG_{3-5}, GAG_{3-5}, AGT_{4-6}} respectively; comparing the codon compositions of 5SSs and 3SSs with the SUFPs in Table 3, it can be seen the identified *ELR*-CCs of SUFP codons closely match current knowledge on sequence patterns of 5SS and 3SS. Moreover, the high-weight SUFPs (highlighted in boldface) such as GTA_{5SS}, AAG_{5SS}, AGT_{5SS}, CAG_{3SS} and TCA_{3SS} also correspond to statistical information in PWMs for 5SS and 3SS consensuses. Accordingly, the validity of the identified patterns is confirmed; furthermore, the specific and detailed information provided by corresponding patterns and are useful annotations that can give deep insights to sequence compositions of splicing signals.

3.4 Discovery of new patterns

In introns, the PPTs are generally characterized as [C/T]-rich regions (about 5∼20 bps upstream of 3SSs' flanks) [Sharp, 1987]; according to the W-CCs of SMFP codons in Table 4, it is clearly specified 7∼17 bps upstream of 3SS is a [TTT/TCT]-rich region; this discovery might represent the core compositions of PPTs; in addition, most of the SMFP codons in classical PPT regions are T-major codons (ID=[22, 24,25,26]); and this consistent with previous finding that continuous Ts are preferred in PPTs [Coolidge et al., 1997]. Besides, some other new patterns in the 5SS downstream and 3SS upstreams are also identified according to the L-CCs and R-CCs; for example, both AAA (ID=[9,27]) and TTT (ID=[14,32]) are rich in the 25∼94 bps downstream of 5SS, and 22∼70 bps upstream of 3SS. In addition, the most significant SMFP codon, GGG(ID=11, weight=1.0), in 5SS downstream is also worthy of notice; the G triplets were identified as intronic splicing enhancers [McCullough and Berget, 1997], which were usually overexpressed in the proximity of intron edges [Majewski and Ott, 2002], the high significance of GGG responds to these findings. Besides, from Table 4, a skewed distribution of GGG (ID=[5,11,17]) to 5SS flank sequences is observed, and this may be a new sequence feature of introns.

3.5 Effectiveness test of the four-faceted sequence patterns

The performance comparisons between the weighted k-NN classifier and un-weighted one are presented. Although no explicit weight vectors were used in the un-weighted k-NN classifier, the Euclidean distance indirectly implied the same importance of all input attributes. Here, we used identity vectors (i.e., all elements in the vector are one) as its weight vectors, and conducted the experiment in the same process as shown in Figure 15 for the performance comparisons.

Since a single performance measure might suffer the risk of being fitted, we carefully used four measures to evaluate the performance; i.e., $error = (fp+fn)/(tp+tn+fp+fn)$, $precision = tp/(tp+fp)$, $recall = tp/(tp+fn)$, and $f1 = 2\times recall\times precision/(recall+precision)$. The factors tp, fp, tn, and fn denote the numbers of predicted true positives, false positives, true negatives, and false negatives, respectively. $Error$ is one of the most used empirical measures that estimate the overall misclassified instances over all instances. *Precision* is a function of the correctly classified examples (tp) and the misclassified examples (fp). *Recall* is a function of tp and fn. Finally; $f1$ measure is an evenly balanced precision and recall. The last three measures distinguish the correct classification of different classes. The reported values of these estimators here are the averages from the 10-fold cross-validation. The performance was shown in the following box-and-whisker diagrams.

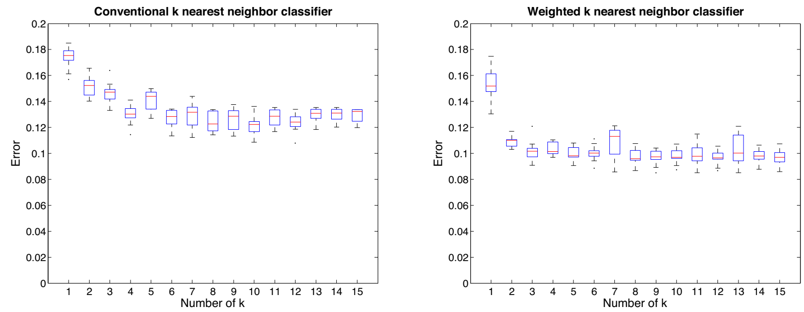

Figure 16: *Error*

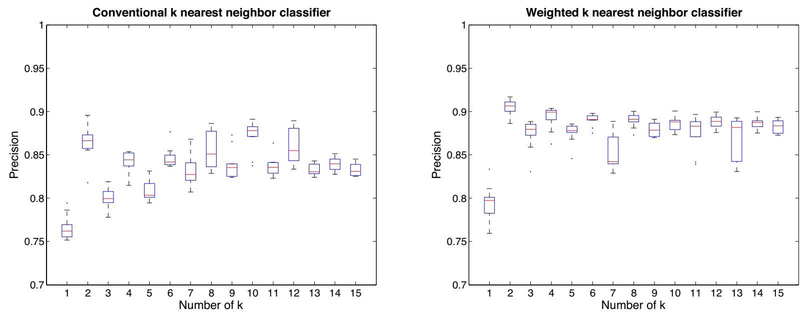

Figure 17: *Precision*

The box-and-whisker diagrams of Figure 16 to Figure 19 clearly indicate that the weighted k-NN classifier performs better than the un-weighted one in terms of *error*, $f1$, and *recall* on different k, except *precision*. In addition to *error* decreased from 2.18% to 6.33%, *precision*, *recall* and $f1$ are also increased 4.168%, 2.03% and 4.61% on average, respectively. From the perspective of k value used in k-NN, slightly better numeric results could be obtained from both weighted and un-weighted nearest neighbor classifiers for $k \geqslant 3$. Furthermore, one might argue that both weighted and conventional k-NN achieve such high scores in precision and relatively low scores in recall; i.e., there are few predicted false positives and lots of predicted false negatives in both models. However, we believe that the reason for this circumstance is due to the inherent model bias and lazy characteristics of the nearest neighbor method. Nearest neighbor classifier is sensitive to the noise because the basic idea is merely distance comparisons. Nevertheless, such a simple weak classifier is appropriate to demonstrate the effectiveness of the patterns.

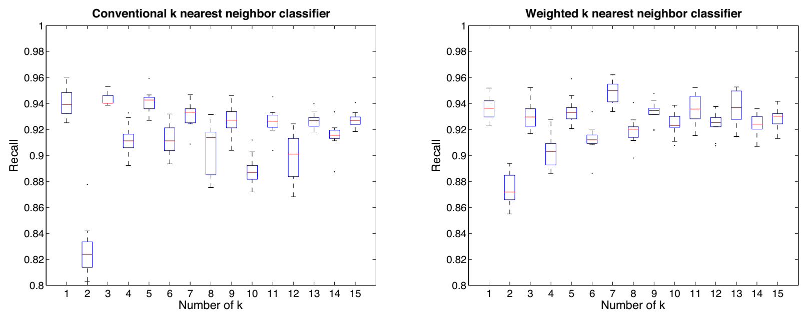

Figure 18: *Recall*

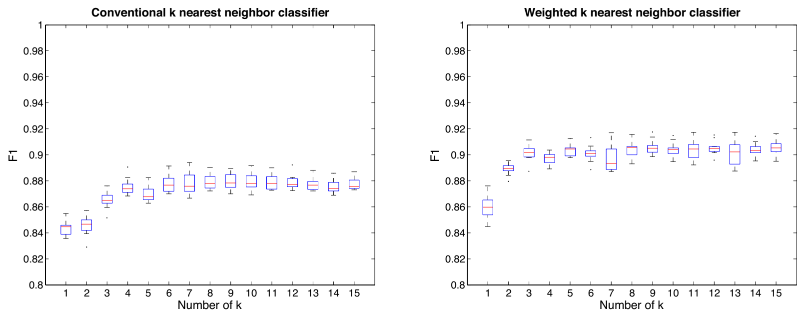

Figure 19: *F1*

Besides, since a limited number of samples were used to compare the performances of two models, we want to know whether the better performance of the weighted *k*-NN classifier is just as a result of the chance effects in the estimation process (i.e., the average estimator of performance measures from 10 folds). More precisely, we should determine whether the observed mean difference of performance measures between two weighted and un-weighted classifiers is really statistically significant. Therefore, we used a paired *t*-test [Montgomery and Runger, 2006] on the weighted *k*-NN classifier and the un-weighted one with a 95% confidence coefficient. In Table 5, the *p*-values over 0.05 are underlined, which reveal that the weight vectors not only significantly reduce the classification error of simple two layered nearest neighbor classifiers, but also significantly improve *recall* and $f1$. In other words, the predicted true positives are enhanced and the false negatives are reduced as well. Thus, we could claim that some meaningful characteristics for intron identification are really enclosed in the patterns. From the selected case study on the intron region, the effectiveness and reliability of

Table 5: The p-values of the t-test on weighted and traditional k-NN classifier

#*k*	Measure			
	error	precision	recall	f1
1	4.68E-03	5.09E-01	2.84E-03	1.29E-02
2	1.37E-07	1.27E-04	6.29E-08	3.56E-04
3	3.31E-07	7.45E-03	2.74E-07	9.25E-07
4	1.69E-05	1.11E-01	3.12E-05	2.16E-05
5	1.70E-06	1.56E-01	1.49E-06	5.50E-06
6	2.43E-05	8.81E-01	5.29E-05	9.91E-06
7	7.89E-04	2.18E-04	4.62E-04	9.19E-03
8	2.91E-06	8.18E-02	2.72E-06	4.97E-04
9	4.00E-06	1.50E-01	3.23E-06	7.86E-05
10	2.23E-05	6.10E-05	5.65E-06	4.90E-02
11	1.17E-06	1.31E-01	8.10E-07	1.12E-04
12	6.33E-06	3.54E-03	1.14E-06	8.02E-03
13	9.47E-05	7.59E-02	5.57E-05	9.88E-04
14	3.20E-08	1.05E-01	1.46E-07	2.98E-08
15	5.68E-08	5.63E-01	1.69E-07	2.94E-09

proposed methods in mining sequence patterns were both demonstrated.

As shown in the results, the small variances (it can be observed from the box-and-whisker diagrams) in each of the measurements, the low error rate among different k and the weighted k-NN outperforming the conventional k-NN; all the above brought us the confidence on the performance of proposed algorithms. The k-NN classifier is appealing because of its simplicity, and the good performance under a simple classifier demonstrates the discriminative power of the mined sequence patterns in distinguishing the true introns from the pseudo ones. Therefore, it is believed the performances obtained via 10-fold cross-validation are un-biased estimations.

4 Discussions

4.1 The *intra*-codon and *int*er-codon analyses

Studies on the three positions of codons had revealed the nucleotide bias in codon usage patterns, the context-dependent codon bias [Fedorov et al., 2002]; and the strong relationship between codon composition and mRNA expressivity had been confirmed [Gouy and Gautier, 1982]. The three codon-positions are also the basis to analyze the base compositional bias between codon positions [Fickett and Tung, 1992]; some patterns of codon usage had been identified, such as the RNY model [Shepherd, 1981, Merino et al., 1992] and the circular code model [Arques and Michel, 1996]. By studying the of the frequencies of three codon-positions, it had been found the distributions of three intron phases is biased in favor of phase 0 [Tomita et al., 1996].

FRPM is a frequency-based analyses model and the fundamental techniques are similar to the researches studying the frequencies of nucleotide in three positions of codons. But the specially devised frequency-counting techniques, such as the *frameshifting, in-frame codon counting* and *codon stepping*, make the FRPM discriminative from these researches that study the frequencies *of in-codon nucleotides*. Studying the frequencies of nucleotide in three positions of codons is *intra-codon* analyses, which focuses on *localization of in-codon nucleotides*; while studying the frequencies in three *frameshift* positions in FRPM is *inter-codon* analyses, which focuses on *tandem repeats* of the codons.

4.2 Flexibility in length requirement of the proposed system

The experimental sequence length was set to be $3k+2$ bps, where k is the number of frames in a sequence; the required length is at least 5 bps (i.e., $k \geq 1$). The computational cost of the proposed method is linearly dependent on the sequence length; thus, it could be said that k has no upper limit. The choice of k mainly depends on the domain knowledge for the studied sequences. In case of intron sequences, the k was set to be 33 in our experiment according to a related pilot study. Currently mined SUFPs and SMFPs (within the 101-bps range) can be reused for shorter introns by a simple modification. Among the proposed four computational concerns, the location (L) and range (R) make the SUFPs and SMFPs be length-adaptable and hence be reusable for shorter introns. While applying the 101-based SUFPs and SMFPs to shorter introns, for all the L-CCs less than the lengths of introns, just reducing the over-length R-CCs to corresponding length limit.

Taking introns as an example, identifying the *short* introns is a special issue in intron-related researches; they are different from those general-length introns in terms of the sequence specificity [Lim and Burge, 2001, Wu et al., 2003]. Therefore, if a certain amount of short introns are available, it would be better to apply the proposed pattern-mining methods to the short-intron data set independently; then comparing the similarities and differences of sequences patterns from the short introns and general-length introns, something valuable might be revealed.

4.3 Potential application in the identification of splice site

For the identification of splice site, the sequence features in both the exonic and intronic flank regions of splice sites is necessary. Based on the proposed pattern-mining methods, a putative model is constructed as shown in Figure 20. For retrieving more specific and detailed sequence patterns flanking splice sites, the regions for mining SUFPs and SMFPs thus cover both the exon and intron flanks. The new set of sequence patterns are believed to be ideal basis for the identification of splice site. By the success in intron identification, it is optimistic that these new *WELR*-based patterns are valuable information for the identification of splice site. However, this model provides only a reliable infrastructure for retrieving sequence features in the proximity of splice site; integrating all the mined sequence patterns to a computational model is truly the key to success in

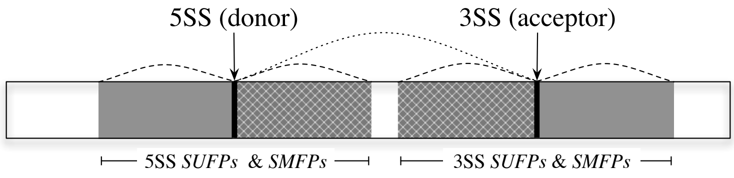

Figure 20: *A putative model for the identification of splice site.*

the identification of splice site, and the techniques to accomplish such a transformation will be varied and domain-knowledge dependent.

5 Conclusions

In this research, a computation-oriented pattern definition model (CO-PDM) was proposed for characterizing the four-faceted sequence patterns; the facets are *weight*, *expression*, *location* and *range*, respectively; each facet is dedicated to be a *computational concern* (CC). A general-purpose system (the GP-MSP) integrated with CO-PDM for mining the four-faceted sequence patterns was devised and implemented; the CO-PDM and the GP-MSP were bridged with the critical-parameter automating (CPA) mechanism. For verifying the effectiveness and reliability of proposed methods, the integrative MSP-mining system was applied to 22,448 human introns as a case study. The well-known intronic sequence patterns were all recovered with more specific and detailed information; moreover, some new patterns were also identified which could provide insights to the pattern deserts in introns. The effectiveness of identified patterns were verified using a two-layered k-NN classifier; both the precision and recall were approximately 90%, therefore, the good performances of GP-MSP and CO-PDM were confirmed. It is believed that the proposed CO-PDM, GP-MSP and the integrated system are all widely applicable for mining patterns in other genomic sequences.

Acknowledgements

This work was supported in part by the National Science Council of Taiwan under the Grant NSC96-2221-E-224-013

References

[Altschul et al., 1994] Altschul, S. F., Boguski, M. S., Gish, W., and Wootton, J. C. (1994). Issues in searching molecular sequence databases. *Nat Genet*, 6(2):119–129.

[Arques and Michel, 1996] Arques, D. G. and Michel, C. J. (1996). A complementary circular code in the protein coding genes. *J Theor Biol*, 182(1):45–58.

[Berg and von Hippel, 1987] Berg, O. G. and von Hippel, P. H. (1987). Selection of dna binding sites by regulatory proteins. statistical-mechanical theory and application to operators and promoters. *J Mol Biol*, 193(4):723–750.

[Brazma et al., 1998] Brazma, A., Jonassen, I., Eidhammer, I., and Gilbert, D. (1998). Approaches to the automatic discovery of patterns in biosequences. *J Comput Biol*, 5(2):279–305.

[Che et al., 2005] Che, D., Jensen, S., Cai, L., and Liu, J. S. (2005). Best: binding-site estimation suite of tools. *Bioinformatics*, 21(12):2909–2911.

[Chen et al., 2007] Chen, X., Guo, L., Fan, Z., and Jiang, T. (2007). Learning position weight matrices from sequence and expression data. *Comput Syst Bioinformatics Conf*, 6:249–260.

[Coolidge et al., 1997] Coolidge, C. J., Seely, R. J., and Patton, J. G. (1997). Functional analysis of the polypyrimidine tract in pre-mrna splicing. *Nucleic Acids Res*, 25(4):888–896.

[Cover and Hart, 1967] Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27.

[Day and McMorris, 1992] Day, W. H. and McMorris, F. R. (1992). Critical comparison of consensus methods for molecular sequences. *Nucleic Acids Res*, 20(5):1093–1099.

[Eberhart and Shi, 1998] Eberhart, R. and Shi, Y. (1998). A modified particle swarm optimizer. In *Proceedings of the IEEE International Conference on Evolutionary Computation*, pages 69–73.

[Fedorov et al., 2002] Fedorov, A., Saxonov, S., and Gilbert, W. (2002). Regularities of context-dependent codon bias in eukaryotic genes. *Nucleic Acids Res*, 30(5):1192–1197.

[Fickett and Tung, 1992] Fickett, J. W. and Tung, C. S. (1992). Assessment of protein coding measures. *Nucleic Acids Res*, 20(24):6441–6450.

[Fratkin et al., 2006] Fratkin, E., Naughton, B. T., Brutlag, D. L., and Batzoglou, S. (2006). Motifcut: regulatory motifs finding with maximum density subgraphs. *Bioinformatics*, 22(14):e150–7.

[Gopalan et al., 2004] Gopalan, V., Tan, T. W., Lee, B. T. K., and Ranganathan, S. (2004). Xpro: database of eukaryotic protein-encoding genes. *Nucleic Acids Res*, 32(Database issue):D59–63.

[Gouy and Gautier, 1982] Gouy, M. and Gautier, C. (1982). Codon usage in bacteria: correlation with gene expressivity. *Nucleic Acids Res*, 10(22):7055–7074.

[Hu et al., 2005] Hu, J., Li, B., and Kihara, D. (2005). Limitations and potentials of current motif discovery algorithms. *Nucleic Acids Res*, 33(15):4899–4913.

[Keich and Pevzner, 2002] Keich, U. and Pevzner, P. A. (2002). Finding motifs in the twilight zone. *Bioinformatics*, 18(10):1374–1381.

[Kennedy and Eberhart, 1995] Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948.

[Lim and Burge, 2001] Lim, L. P. and Burge, C. B. (2001). A computational analysis of sequence features involved in recognition of short introns. *Proc Natl Acad Sci U S A*, 98(20):11193–11198.

[Liu et al., 2001] Liu, X., Brutlag, D. L., and Liu, J. S. (2001). Bioprospector: discovering conserved dna motifs in upstream regulatory regions of co-expressed genes. *Pac Symp Biocomput*, pages 127–138.

[MacIsaac and Fraenkel, 2006] MacIsaac, K. D. and Fraenkel, E. (2006). Practical strategies for discovering regulatory dna sequence motifs. *PLoS Comput Biol*, 2(4):e36.

[Majewski and Ott, 2002] Majewski, J. and Ott, J. (2002). Distribution and characterization of regulatory elements in the human genome. *Genome Res*, 12(12):1827–1836.

[McCullough and Berget, 1997] McCullough, A. J. and Berget, S. M. (1997). G triplets located throughout a class of small vertebrate introns enforce intron borders and regulate splice site selection. *Mol Cell Biol*, 17(8):4562–4571.

[Merino et al., 1992] Merino, E., Balbas, P., and Bolivar, F. (1992). New insights on the commaless theory. *Orig Life Evol Biosph*, 21(4):251–254.

[Mitchell, 1997] Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York.

[Montgomery and Runger, 2006] Montgomery, D. C. and Runger, G. C. (2006). *Applied Statistics and Probability for Engineers*. Wiley.

[Moore, 2000] Moore, M. J. (2000). Intron recognition comes of age. *Nat Struct Biol*, 7(1):14–16.

[Patel and Steitz, 2003] Patel, A. A. and Steitz, J. A. (2003). Splicing double: insights from the second spliceosome. *Nat Rev Mol Cell Biol*, 4(12):960–970.

[Pribnow, 1975] Pribnow, D. (1975). Nucleotide sequence of an rna polymerase binding site at an early t7 promoter. *Proc Natl Acad Sci U S A*, 72(3):784–788.

[Reddy et al., 2006] Reddy, C. K., Weng, Y.-C., and Chiang, H.-D. (2006). Refining motifs by improving information content scores using neighborhood profile search. *Algorithms Mol Biol*, 1:23.

[Roth et al., 1998] Roth, F. P., Hughes, J. D., Estep, P. W., and Church, G. M. (1998). Finding dna regulatory motifs within unaligned noncoding sequences clustered by whole-genome mrna quantitation. *Nat Biotechnol*, 16(10):939–945.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representations by error propagation. pages 318–362.

[Schneider et al., 1986] Schneider, T. D., Stormo, G. D., Gold, L., and Ehrenfeucht, A. (1986). Information content of binding sites on nucleotide sequences. *J Mol Biol*, 188(3):415–431.

[Sharp, 1987] Sharp, P. A. (1987). Splicing of messenger rna precursors. *Science*, 235(4790):766–771.

[Shepherd, 1981] Shepherd, J. C. (1981). Method to determine the reading frame of a protein from the purine/pyrimidine genome sequence and its possible evolutionary justification. *Proc Natl Acad Sci U S A*, 78(3):1596–1600.

[Sinha, 2006] Sinha, S. (2006). On counting position weight matrix matches in a sequence, with application to discriminative motif finding. *Bioinformatics*, 22(14):e454–63.

[Steiger and Sharda, 1996] Steiger, D. M. and Sharda, R. (1996). Analyzing mathematical models with inductive learning networks. *European Journal of Operational Research*, 93(2):387–401.

[Stormo, 2000] Stormo, G. D. (2000). Dna binding sites: representation and discovery. *Bioinformatics*, 16(1):16–23.

[Stormo et al., 1982] Stormo, G. D., Schneider, T. D., and Gold, L. M. (1982). Characterization of translational initiation sites in e. coli. *Nucleic Acids Res*, 10(9):2971–2996.

[Tomita et al., 1996] Tomita, M., Shimizu, N., and Brutlag, D. L. (1996). Introns and reading frames: correlation between splicing sites and their codon positions. *Mol Biol Evol*, 13(9):1219–1223.

[Werbos, 1974] Werbos, P. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD thesis, Harvard University, Cambridge, MA.

[Wu et al., 2003] Wu, Y., Liew, A. W.-C., Yan, H., and Yang, M. (2003). Classification of short human exons and introns based on statistical features. *Phys Rev E Stat Nonlin Soft Matter Phys*, 67(6 Pt 1):061916.

[Yao et al., 2006] Yao, Z., Weinberg, Z., and Ruzzo, W. L. (2006). Cmfinder–a covariance model based rna motif finding algorithm. *Bioinformatics*, 22(4):445–452.

[Yoon et al., 1994] Yoon, Y., Guimaraes, T., and Swales, G. (1994). Integrating artificial neural networks with rule-based expert systems. *Decision Support Systems*.