Polyadenylation Site Prediction Using Interesting Emerging Patterns

George Tzanis, *Member, IEEE*, Ioannis Kavakiotis, and Ioannis Vlahavas, *Member, IEEE*

***Abstract*—This paper presents a study on polyadenylation site prediction in mRNA sequences. We describe a method, called PolyA-EP, that we developed for predicting polyadenylation sites and we present a systematic study of the problem of recognizing mRNA 3′ ends which contain a polyadenylation site using the proposed method. PolyA-EP exploits the advantages of emerging patterns, namely high understandability and discriminating power and can be used for both descriptive and predictive analysis. In particular, PolyA-EP is a parameterizable tool that can be used in order to extract interesting emerging patterns for describing or predicting polyadenylation sites. Moreover, the extracted emerging patterns can span across many elements around the polyadenylation site. We discuss the results of the experiments we conducted with Arabidopsis thaliana sequences drawing important conclusions and finally we propose a framework that improves the accuracy of polyadenylation site prediction.**

I. Introduction

POLYADENYLATION is a process that occurs after transcription termination. It involves cleavage of the new transcript (mRNA), followed by template-independent addition of adenines at its newly synthesized 3′ end. The cleavage site is called polyadenylation site (poly(A) site). Polyadenylation is considered to be part of the larger process of producing mature mRNA for translation. The aim of the polyadenylation process is to protect the mRNA in order to reach intact the protein synthesis site.

The most important factors that are involved in the process of polyadenylation are the cis-regulatory elements and the trans-acting factors. The cis-regulatory elements are RNA sequences consisting from 2 to 10 nucleotides and their role is to help the trans-action factors define the poly(A) site. The most prominent cis-element is the hexamer AAUAAA or a close variant. This hexamer is located 10 – 35 nt upstream of the cleavage site (poly(A)-site) and it can be found in about 50% of human genes [5] but only in 10% of Arabidopsis genes [8]. The trans-acting factors are a protein complex which also includes a specificity factor (Cleavage and Polyadenylation Specificity Factor - CPSF), an endonuclease, and Poly(A) Polymerase (PAP). The trans-acting factors are responsible for the cleavage at the appropriate site (poly(A) site) and the addition of the about 200 adenine residues (poly(A) tail) to the 3′ end [11].

Nowadays, the research in this field is focused on discovering new cis-regulatory elements and on predicting the poly(A) site accurately. The accurate prediction of poly(A) site is a crucial step to define gene boundaries and get an insight in transcription termination in eukaryotes, which is a process less well understood.

Poly(A) site prediction is a challenging problem. In many organisms, such as in Arabidopsis thaliana there are not many highly conserved signals or patterns around the poly(A) site and consequently the recognition of the poly(A) site is not trivial. The discrimination of mRNA 3′ ends that contain a poly(A) site from intronic or 5′ UTR sequences without a poly(A) site seems to be very difficult (especially with intronic sequences) and the performance of the up to now proposed approaches is moderate. On the other hand, mRNA 3′ ends can be easily discriminated from coding sequences. This variability in the difficulty of discrimination has motivated our work and guided us to an effort to study this problem and define an approach that can improve prediction accuracy.

Our contribution is a method that exploits the twofold advantage of emerging patterns, namely their high interpretability and discriminating power. The method we propose can be parameterized and trained in order to deal with poly(A) site prediction in any organism. Beyond the proposed method we draw important conclusions on the problem of discriminating mRNA 3′ ends with poly(A) sites from other sequences without a poly(A) site.

II. Related Work

An early approach to the problem of poly(A) site prediction was the work of A.A. Salamov and V.V. Solovyev [9] who developed a software called POLYAH and an algorithm for the identification of 3′-processing sites of human mRNA precursors. The algorithm was based on a linear discriminant function (LDF) trained to discriminate real poly(A) signals from the other regions of human genes possessing the AATAAA sequence which is most likely non functional. The accuracy of the method has been estimated on a set of 131 poly(A) regions and 1466 regions of human genes having the AATAAA sequence. When the threshold was set to predict 86% of poly(A) regions correctly, specificity of 51% and correlation coefficient of 0.62 had

Manuscript received July 5, 2008.

George Tzanis is with the Department of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece (corresponding author: +302310998433; e-mail: gtzanis@csd.auth.gr).

Ioannis Kavakiotis is with the Department of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece (e-mail: ikavak@csd.auth.gr).

Ioannis Vlahavas is with the Department of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece (+302310998145; e-mail: vlahavas@csd.auth.gr).

been achieved.

In 1999 Tabaska and Zhang [10] developed polyadq, a program for detection of human polyadenylation signals. The program finds poly(A) signals using two discriminant functions: one specific for AATAAA type poly(A) sites and the other for ATTAAA type poly(A) sites. Polyadq predicts poly(A) signals with a correlation coefficient of 0.413 on whole genes and 0.512 in the last two exons of genes.

In 2000 Van Helden et al. [4] approached the poly(A) site prediction problem with statistical methods. Other interesting approaches on this problem was the Hidden Markov Model approaches by Graber et al. [2] and Hajarnavis et al. [3].

In 2003 Liu et al. [7] proposed a machine learning method to predict polyadenylation signals in human RNA sequences by analyzing features around them. The method consists of three steps: (1) Generating candidate features from the original sequence data using k-gram nucleotide patterns or amino acid patterns. (2) Selecting relevant features using an entropy-based algorithm. (3) Integrating the selected features by SVMs to build a system to recognize poly(A) sites.

In 2005 Hu et al. [5] developed a program named PROBE (Polyadenylation-Related Oligonucleotide Bidimensional Enrichment) to identify cis-elements that may play regulatory roles in mRNA polyadenylation. They found 15 cis-elements in the area of 100 nt upstream and downstream the poly(A) site. Another important conclusion of this work was that cis-elements occurring in yeast and plants also exist in human poly(A) regions. They suggested that many cis-elements are evolutionarily conserved among eukaryotes and human poly(A) sites have an additional set of cis elements that may be involved in the regulation of mRNA polyadenylation.

A year later Cheng et al. [1] from the same lab tried to address whether those 15 cis-elements could be used to predict poly(A) sites. So they developed a program called Polya_svm which used support vector machines in order to predict poly(A) sites exploiting these 15 cis-elements. Polya_svm achieved higher sensitivity and similar specificity when compared with polyadq.

One of the most recent projects in the scientific area of polyadenylation site prediction was published in 2007 by Ji et al. [6]. Ji and his co-workers exploited the conclusions of Loke's study [8] and developed a program named PASS (Poly(A) site sleuth) which used a Generalized Hidden Markov Model based algorithm in order to predict polyadenylation sites in Arabidopsis. Additionally, researchers from the same lab recently published a work in which they developed a program called Pass-Rice and predicts poly(A) sites in rice data [13].

Another approach to the poly(A) site prediction problem was made by C. Koh and L. Wong [12]. Their prediction model uses a machine learning approach which consists of four sequential steps: feature generation, feature selection, feature integration and a cascade support vector machine classifier.

III. Preliminaries

A. Frequent Itemsets

The term "frequent itemset" has been proposed in the framework of association rules mining. Association rules [14] have attracted the attention of the data mining research community since the early 90s, as a means of unsupervised, exploratory data analysis. The association rule mining paradigm involves searching for co-occurrences of items in transaction databases. Such a co-occurrence may imply a relationship among the items it associates. The task of mining association rules consists of two main steps. The first one includes the discovery of all the frequent itemsets contained in a transaction database. In the second step, the association rules are generated from the discovered frequent itemsets. A formal statement of the concept of frequent itemsets is presented in the following paragraph.

Let $I = \{i_1, i_2, \ldots, i_N\}$ be a finite set of binary attributes which are called *items* and D be a finite multiset of *transactions*, which is called *dataset*. Each transaction $T \in D$ is a set of items such that $T \subseteq I$. A set of items is usually called an *itemset*. The *length* or *size* of an itemset is the number of items it contains. It is said that a transaction $T \in D$ *contains* an itemset $X \subseteq I$, if $X \subseteq T$. The *support* of itemset X is defined as the fraction of the transactions that contain itemset X over the total number of transactions in D:

$$supp_D(X) = \frac{\left|\{T \in D \mid T \supseteq X\}\right|}{|D|} \tag{1}$$

Given a minimum support threshold $\sigma \in (0,1]$, an itemset X is said to be *σ-frequent*, or simply *frequent* in D, if $supp_D(X) \geq \sigma$.

B. Emerging Patterns

Emerging patterns [16] are itemsets whose supports increase significantly from one dataset to another.

Given two datasets D_1 and D_2, the *growth rate* of an itemset X from D_1 to D_2 is defined as (indices 1 and 2 are used instead of D_1 and D_2):

$$gr_{1\to 2}(X) = \begin{cases} 0, & \text{if } supp_1(X) = 0 \text{ and } supp_2(X) = 0 \\ \infty, & \text{if } supp_1(X) = 0 \text{ and } supp_2(X) > 0 \\ \dfrac{supp_2(X)}{supp_1(X)}, & \text{otherwise} \end{cases} \tag{2}$$

Given a minimum growth rate threshold $\rho > 1$, an itemset X is said to be *ρ-emerging pattern*, or simply *emerging pattern*, from D_1 to D_2, if $gr_{1\to 2}(X) \geq \rho$. D_1 is usually called *background dataset* and D_2 is usually called *target dataset*.

The *strength* of an emerging pattern X from D_1 to D_2 is defined as:

$$strength_{1\to 2}(X) = \begin{cases} supp_2(X), & \text{if } gr_{1\to 2}(X) = \infty \\ supp_2(X)\dfrac{gr_{1\to 2}(X)}{gr_{1\to 2}(X)+1}, & \text{otherwise} \end{cases} \quad (3)$$

Emerging patterns in contrast to other patterns or models are easily interpretable and understood. Moreover, emerging patterns, especially those with a large growth rate and strength, provide a great potential for discriminating examples of different classes. This twofold benefit of emerging patterns makes them a useful tool for exploring domains that are not well understood, providing the means for descriptive and predictive analysis as well.

However, a disadvantage of emerging pattern mining is that the number of emerging patterns may be huge, especially when minimum support and minimum growth rate thresholds are set very low. Increasing the thresholds is not an ideal solution, since valuable emerging patterns may not be discovered. For example, if minimum support threshold is set high, then those emerging patterns with a low support, but with a high growth rate will be lost. Conversely, if minimum growth rate threshold is set high, then those emerging patterns with a low growth rate, but with a high support will be lost. There have been proposed some interestingness measures in order to reduce the number of mined emerging patterns without sacrificing valuable emerging patterns, or at least sacrificing as less as possible. Such an interestingness measure includes a special kind of emerging patterns, called *Chi Emerging Patterns* [18], that are defined as follows.

Given a background dataset D_1 and a target dataset D_2, an itemset X is called a chi emerging pattern if all the following conditions are true:

1) $supp_2(X) \geq \sigma$, where σ is a minimum support threshold.
2) $gr_{1\to 2}(X) \geq \rho$, where ρ is a minimum growth rate threshold.
3) $\forall Y \subset X, gr_{1\to 2}(Y) < gr_{1\to 2}(X)$
4) $|X| = 1 \vee |X| > 1 \wedge (\forall Y \subset X \wedge |Y| = |X|\text{-}1 \wedge chi(X,Y) \geq \eta)$, where η = 3.84 is a minimum chi value threshold and $chi(X, Y)$ is computed using chi-squared test.

The first condition ensures that the mined emerging patterns will have at least a minimum coverage over the training dataset in order to generalize well on new instances. The second condition ensures that the mined emerging patterns will have an adequate discriminating power. The third condition is used in order to filter out those emerging patterns that have a subset with higher or equal growth rate and higher or equal support (any itemset has equal or greater support than any of its supersets). Since the subset has fewer items, there is not any reason to keep this emerging pattern. Finally, the fourth condition ensures that an emerging pattern has a significantly (95%) different support distribution in target and background datasets than the distributions of its immediate subsets.

IV. Our Method

In this paragraph we describe the method (PolyA-EP) we have developed for dealing with the problem of polyadenylation site prediction in Arabidopsis thaliana mRNA sequences. Although in this study we have concentrated on a plant that poses great challenges due to low conservation of poly(A) signals, the method we propose is abstract and can be re-trained and parameterized for studying different organisms. PolyA-EP has been implemented in JAVA and consists of a number of steps that are presented in detail below.

A. Extraction of Elements

There is a number of different elements around the cleavage site of an mRNA 3′ end that have been recognized in previous studies (Figure 1). These elements are composed by different nucleotide frequencies and consequently may contain fairly different patterns. This indicates that one has to search for patterns separately in each element. However, a promising idea is to study the associations among the patterns of the different elements in order to discover possible relationships among them. This could lead to new "extended" patterns that are possibly more informative and have higher discriminating power than the single patterns found in each element separately. In our study we deal with this kind of "extended" patterns. The three basic elements located around Arabidopsis 3′ end poly(A) sites have been proposed in previous studies (see for example [8]) and include the Far Upstream Element (FUE), the Near Upstream Element (NUE) and the Cleavage Element (CE). The downstream region of Arabidopsis poly(A) sites is not considered particularly important, however we have included a Near Downstream Element (NDE) in our study, in order to investigate its importance.

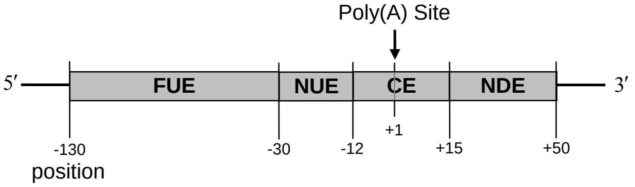

Fig. 1. A model for poly(A) signals in Arabidopsis mRNA 3′ ends.

At the first step of our method the elements specified by the user are extracted. In our study we have used the elements that are presented in Figure 1.

B. Extraction of k-Grams

Each of the sequence elements that are extracted at the first step will be represented by a vector that contains the frequencies of 5460 nucleotide patterns (k-grams). These patterns include all nucleotide combinations of length k, where $k \in \{1,2,...,6\}$. So, each initial sequence from now on will be represented by a number of vectors, one for each of the specified elements (i.e. FUE, NUE, CE, NDE). The user of PolyA-EP can specify a different k.

C. Binary Discretization

The discretization method used in our approach is based on information entropy. For each *k*-gram pattern a cut point is sought among all pattern frequencies and the one that has the maximum information gain is finally selected. The *k*-gram vectors that were previously constructed are transformed into a transaction of items. The items of the transaction are those *k*-grams that have a frequency value greater than the corresponding cut point, which has been calculated. In this step the data are transformed in a format that permits the extraction of emerging patterns.

D. Mining Interesting Emerging Patterns

At this step the transactional data that have been produced in the previous step can be mined for interesting emerging patterns. For this reason we have extended the FP-Growth algorithm [15] that is used for mining frequent itemsets. The extended algorithm receives as input two datasets, the background and the target dataset, and discovers all chi emerging patterns, based on the parameters specified by the user (minimum support threshold and minimum growth rate threshold). At this point it is worthwhile to repeat that the patterns that are mined by PolyA-EP are "extended", since these patterns can include itemsets of different elements.

E. Classification Using Interesting Emerging Patterns

The extracted chi emerging patterns can be used in order to discriminate instances of different classes. Given two datasets of sequences D_+ and D_- that contain sequences with a poly(A) site (positive sequences) and sequences without a poly(A) site (negative sequences) respectively, two sets of emerging patterns E_+ and E_- are mined. For mining E_+, D_- will be the background dataset and D_+ will be the target dataset. In contrast for mining E_-, D_+ will be used as the background dataset and D_- as the target dataset. When a new instance has to be classified it is transformed in a transaction T as described previously. Then, the following scores are calculated.

$$\begin{aligned} score(T,+) &= \sum_{e \subseteq T, e \in E_+} strength_{-\rightarrow +}(e) \\ score(T,-) &= \sum_{e \subseteq T, e \in E_-} strength_{+\rightarrow -}(e) \end{aligned} \qquad (4)$$

The first score indicates if T is positive and the second if it is negative. The final decision could be made by comparing the values of the two scores and assigning the sequence to the class with the higher score. However, due to the fact that the sizes of E_+ and E_- could be quite different the scores have to be justified. We have studied three alternative methods:

1) The first method was presented in [17]. It calculates two base scores, *base*$_+$ and *base*$_-$, for positive and negative classes respectively. The *base*$_+$ score is found by calculating the positive score (using E_+) for each of the instances of the positive training set, and selecting the median of the scores to be *base*$_+$. Similarly is calculated *base*-, using negative training instances and E_- instead. The two scores that are calculated for a new instance are divided by the corresponding base scores and the instance is finally assigned to the class with the greatest justified score.
2) We studied the use of information entropy in order to select a threshold for the following fraction $\frac{score(T,+)}{score(T,-)}$. This fraction is calculated for all of the training (positive and negative) instances and a cut point, *entropy_thres*, which maximizes information gain is found. A new instance is assigned to positive class if the above fraction exceeds *entropy_thres*.
3) We studied a combination of the above two score justification methods and propose another threshold for the fraction in 2. This threshold is defined as follows:

$$entropy\_base = \frac{entropy\_thres + \frac{base_+}{base_-}}{2} \qquad (5)$$

The experiments we have conducted (not presented here due to space limitations) indicated that the justification method presented in 1 tends to favor the class with the smallest number of training instances, whereas the method in 2 tends to favor the class with the majority of training instances. For this reason we propose the score justification method in 3 that balances the previous two methods.

V. Experiments

In this section we describe the datasets we have used as well as the experiments we have conducted in order to evaluate our method.

A. Datasets

In our study we have used four sets of Arabidopsis thaliana sequences. One of them contains 6209 positive examples, namely mRNA 3′ end sequences that contain a poly(A) site, whereas the other three contain negative examples (1581 intronic, 864 5′ UTR, and 1501 coding sequences). These data have been used in previous studies [6], [12]. The set of positive sequences will be called positive dataset and the set of all negative sequences will be called negative dataset. All sequences have a length of 400 nt. Each positive sequence has an EST-supported poly(A) site at position 301. The positive sequences underwent pair-wise global alignment against every other sequence [12] in order to reduce similarity among all sequences. Particularly, there are not any two sequences in the positive dataset that have more than 70% similarity. This was done for minimizing biasness due to similarity of sequences. More details about these datasets can be found in [6].

B. Results

We have conducted a number of experiments using the above datasets. For evaluating our method, we have randomly selected 2/3 of each of the four sets of sequences for training and left the remaining sequences for testing.

Table I presents the experimental results of mining chi emerging patterns using all the training negative examples (i.e. intronic, 5′ UTR and coding) together. As shown in the table a lower minimum support threshold and a lower minimum growth rate threshold result a larger number of mined chi emerging patterns. This is expected, but it is not always the case. If simple emerging patterns are mined, then lower thresholds will always lead to equal or larger number of mined emerging patterns. Because of conditions 3 and 4 in the definition of chi emerging patterns, it is not certain that lower thresholds guide to equal or greater number of chi emerging patterns. An important conclusion that can be drawn based on Table I is that when the number of mined emerging patterns is greater, then higher classification accuracy is achieved. However, experiments have shown that a number of at least 1000 chi emerging patterns for each class are adequate for getting good classification performance. It is worthwhile to mention that in contrast to simple emerging patterns, chi emerging patterns are not redundant and thus a larger number of chi emerging patterns almost always improves classification accuracy.

TABLE I
RESULTS OF MINING INTERESTING EMERGING PATTERNS USING TRAINING POSITIVE AND ALL TRAINING NEGATIVE SEQUENCES

support threshold	growth rate threshold	# positive chi EPs	# negative chi EPs	sensitivity	specificity
0.1	2	25715	4752	0.875	0.750
	5	6135	1323	0.878	0.747
0.2	2	5755	406	0.859	0.738
	5	533	9	0.861	0.678

Table II presents the number of chi emerging patterns that were mined using all the training positive examples and the three negative datasets separately. The minimum support threshold was set to 0.1 and the minimum growth rate varied between 2 and 5. As shown in the table, this support threshold is too high for mining an adequate number of chi emerging patterns with the intronic negative dataset. However, it guides to a very large number of chi emerging patterns with the coding negative dataset. Support threshold is also high for 5′ UTRs, but not as high is for introns. Table II provides useful information that could not be discovered when all negative examples were dealt together. The important conclusion is that one cannot adequately discriminate positive sequences from intronic and 5′ UTR sequences. Moreover, we can also conclude that the problem of discriminating introns and 5′ UTRs from mRNA 3′ ends is quite difficult, whereas discriminating mRNA 3′ ends from coding sequences is much easier.

TABLE II
NUMBER OF INTERESTING EMERGING PATTERNS USING THE THREE NEGATIVE DATASETS SEPARATELY (MINIMUM SUPPORT THRESHOLD = 0.1)

negative dataset	growth rate threshold	# positive chi EPs	# negative chi EPs
intronic	2	50	431
	5	21	1
5′ UTR	2	409	2946
	5	300	760
coding	2	32623	17996
	5	32377	31162

In an effort to further investigate why the number of chi emerging patterns differs so much among the three negative datasets we plotted the distributions of nucleotides for each dataset. Figure 2 presents the distribution of each nucleotide from positions -200 to +100 with respect to the poly(A) site in Arabidopsis mRNA 3′ ends. The differences in nucleotide distributions among different elements are very clear. Figures 3, 4, and 5 depict the nucleotide distributions of intronic, 5′ UTR, and coding Arabidopsis sequences. These figures clearly depict why the discrimination between positive sequences and negative intronic sequences is very difficult, whereas the discrimination between positive and negative coding sequences is easy. Comparing figures 2 and 3, we can see that there are many similarities in nucleotide distributions of mRNA 3′ end sequences and introns. In intronic sequences, uracil is the most frequent nucleotide, followed by adenine, then guanine, and finally cytosine. This is also the case with the upstream region up to the NUE of the mRNA 3′ end. In contrast, the nucleotide distribution of coding sequences is very different than the one of mRNA 3′ ends. Finally, 5′ UTR sequences have also similar nucleotide distribution with this of introns, but they differ from mRNA 3′ ends more than introns do. That is the reason, why 5′ UTRs can be discriminated easier from mRNA 3′ ends.

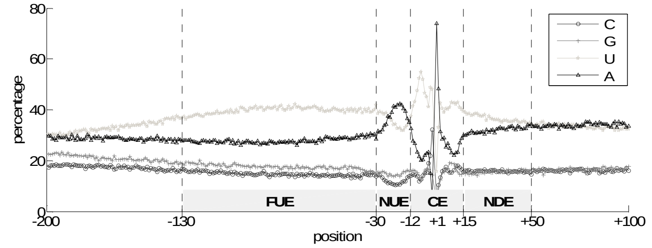

Fig. 2. Nucleotide distribution from positions -200 to +100 with respect to poly(A) site in Arabidopsis mRNA 3′ end.

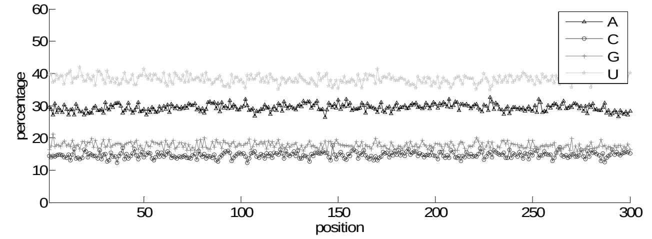

Fig. 3. Nucleotide distribution in Arabidopsis intronic sequences.

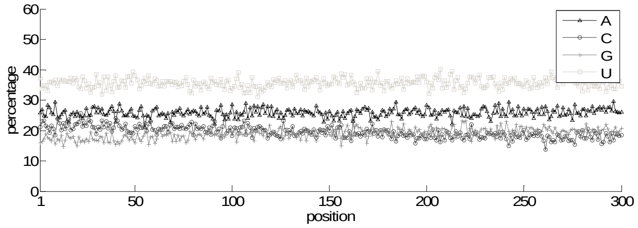

Fig. 4. Nucleotide distribution in Arabidopsis 5´ UTR sequences.

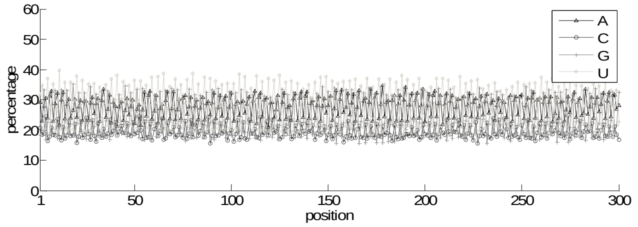

Fig. 5. Nucleotide distribution in Arabidopsis coding sequences.

The results presented previously guided us to deal with the three negative datasets separately. Table III shows the results of mining chi emerging patterns with minimum growth rate of 2 and different minimum support thresholds for each of the three cases. If the minimum support threshold is set to 0.02, a relatively large number of chi emerging patterns is mined in the case of intronic data. However, when the minimum support threshold is lowered too much, the risk of mining chi emerging patterns that over-fit the training data emerges and consequently the generalization error increases. However, dealing with the three negative datasets separately improved the overall classification performance (sensitivity: 0.891 and specificity: 0.874).

TABLE III
NUMBER OF INTERESTING EMERGING PATTERNS USING THREE NEGATIVE DATASETS SEPARATELY (MINIMUM GROWTH RATE THRESHOLD = 2)

negative dataset	support threshold	# positive chi EPs	# negative chi EPs
intronic	0.02	2543	10073
5´ UTR	0.05	2144	12647
coding	0.1	32623	17996

VI. CONCLUSION

Polyadenylation site prediction is a challenging problem that has not yet been sufficiently dealt. Nowadays, the research in this field is focused on discovering new cis-regulatory elements and on predicting the poly(A) site accurately. The difficulties on poly(A) site prediction are basically derived by the absence of highly conserved signals around the poly(A) site. In this work we studied the problem of poly(A) site prediction and proposed a method (PolyA-EP) that can be used for both descriptive and predictive analysis. PolyA-EP exploits emerging patterns and eventually provides a framework for increasing prediction accuracy.

In the future we are considering to incorporate in our method patterns of larger lengths (in this work we used patterns of length 1 to 6), as well as to allow the patterns to include wildcards. However, in these cases we will have to deal with the high computational cost. But the incorporation of such patterns is considered to provide more interesting emerging patterns with higher discriminative power, something that is quite important, especially in the hard problem of discriminating mRNA 3´ ends from intronic sequences. Finally, our future plans include the experimentation with mRNA sequences of other organisms.

The datasets we used and the tool we developed are available at http://mlkd.csd.auth.gr/PolyA/index.html.

REFERENCES

[1] Y. Cheng, R.M. Miura, and B. Tian, “Prediction of mRNA polyadenylation sites by support vector machine”. In *Bioinformatics* 2006, 22(19), pp. 2320-2325.

[2] J.H. Graber, G.D. McAllister, and T.F. Smith, “Probabilistic prediction of Saccharomyces cerevisiae mRNA 3'-processing sites”. In *Nucleic Acids Research* 2002, 30(8), pp. 1851-1858.

[3] A. Hajarnavis, I. Korf, and R. Durbin, “A probabilistic model of 30 end formation in Caenorhabditis elegans”. In *Nucleic Acids Research* 2004, 32, pp. 3392–3399.

[4] J. Van Helden, M del Olmo, and J.E. Perez-Ortin, “Statistical analysis of yeast genomic downsream sequences reveals putative polyadenylation signals”. In *Nucleic Acids Research*, 28, 1000-1010.

[5] J Hu, C.S. Lutz, J. Wilusz, and B. Tian, “Bioinformatic identification ofcandidate cis-regulatory elements involved in human mRNA polyadenylation”. In *RNA* 2005, 11(10) pp. 1485-1493.

[6] G. Ji, J. Zheng, Y. Shen, X. Wu, R. Jiang, Y. Lin, J. Loke, K, Davis, G. Reese, and Q. Li, “Predictive modeling of plant messenger RNA polyadenylation sites”. In *BMC Bioinformatics*, 2007, 8:43.

[7] H. Liu, H. Han, J. Li, and L. Wong, “An in-silico method for prediction of polyadenylation signals in human sequences”. In *Genome Inform Ser Workshop Genome Inform* 2003, 14, pp. 84-93.

[8] J. Loke, E.A. Stahlberg, D.G. Strenski, B.J. Haas, P.C. Wood, and Q.Q. Li, “Compilation of mRNA polyadenylation signals in Arabidopsis revealed a new signal element and potential secondary structures. In *Plant Physiol*ogy 2005, 138, pp. 1457-1468.

[9] A. Salamov and V. Solovyev, “Recognition of 30-processing sites of human mRNA precursors”. In *Comput. Appl. Biosci*. 1997, 13, 23-28.

[10] J.E. Tabaska and M.Q. Zhang, “Detection of polyadenylation signals in human DNA sequences”. In *Gene* 1999, 231, pp. 77–86.

[11] B. Lewin, *Genes VIII*. Pearson Education Inc. 2004 pp. 721-722.

[12] C.H. Koh, and L. Wong. ‘Recognition of polyadenylation sites from Arabidopsis genomic sequenses”. In *Proceedings of 18th International Conference on Genome Informatics*, pp. 73-82, 2007.

[13] Y. Shen, G. Ji, B.J. Haas, X. Wu, J. Zheng, G.J. Reese, and Q.Q. Li, “Genome level analysis of rice mRNA 3’-end processing signals and alternative polyadenylation”. In *Nucleic Acids Research*, 36(9), pp. 3150-3161.

[14] R. Agrawal, T. Imielinski, and A. Swami, “Mining association rules between sets of items in large databases”. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, 1993, pp. 207-216.

[15] J. Han, J. Pei, and Y. Yin. “Mining frequent patterns without candidate generation”. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, 2000, pp. 1-12.

[16] G. Dong and J. Li, “Efficient mining of emerging patterns: Discovering trends and differences”. In *Proceedings of ACM-SIGKDD’99*, 1999, pp. 43–52.

[17] G. Dong, X. Zhang, L. Wong, and J. Li, “CAEP: Classification by aggregating emerging patterns”. *In Proceedings the 2nd International Conference on Discovery Science*, 1999, pp. 30–42.

[18] H. Fan. *Efficient Mining of Interesting Emerging Patterns and Their Effective Use in Classification*, PhD Thesis, University of Melbourne, Australia, 2004.