A Feature Generation Algorithm for Sequences with Application to Splice-Site Prediction

Rezarta Islamaj[1], Lise Getoor[1], and W. John Wilbur[2]

[1] Computer Science Department, University of Maryland, College Park, MD 20742
[2] National Center for Biotechnology Information, NLM, NIH, Bethesda, MD 20894
{rezarta, getoor}@cs.umd.edu, wilbur@ncbi.nlm.nih.gov

Abstract. In this paper we present a new approach to feature selection for sequence data. We identify general feature categories and give construction algorithms for them. We show how they can be integrated in a system that tightly couples feature construction and feature selection. This integrated process, which we refer to as *feature generation*, allows us to systematically search a large space of potential features. We demonstrate the effectiveness of our approach for an important component of the gene finding problem, splice-site prediction. We show that predictive models built using our feature generation algorithm achieve a significant improvement in accuracy over existing, state-of-the-art approaches.

Keywords: feature generation, splice-site prediction.

1 Introduction

Many real-world data mining problems involve data modeled as sequences. Sequence data comes in many forms including: 1) human communication such as speech, handwriting and language, 2) time sequences and sensor readings such as stock market prices, temperature readings and web-click streams and 3) biological sequences such as DNA, RNA and protein. In all these domains it is important to efficiently identify useful 'signals' in the data that enable the correct construction of classification algorithms.

Extracting and interpreting these 'signals' is known to be a hard problem. The focus of this paper is on a systematic and scalable method for feature generation for sequences. We identify a collection of generic sequence feature types and describe the corresponding feature construction methods. These methods can be used to create more complex feature representations. As exhaustive search of this large space of potential features is intractable, we propose a general-purpose, focused **feature generation algorithm (FGA)**, which integrates feature construction and feature selection. The output of the feature generation algorithm is a moderately sized set of features which can be used by arbitrary classification algorithm to build a classifier for sequence prediction.

We validate our method on the task of splice-site prediction for pre-mRNA sequences. Splice sites are locations in the DNA sequence which are boundaries for protein coding regions and non-coding regions. Accurate prediction of splice sites is an important component of the gene finding problem. It is a particularly difficult problem since the sequence characteristics, e.g. pre-mRNA sequence length, coding sequence length, number of interrupting intron sequences and their lengths, do not follow any known pattern, making it hard to locate the genes. The gene finding challenge is to build a general approach that, despite the lack of known patterns, will automatically select the right features to combine.

We demonstrate the effectiveness of this approach by comparing it with a state-of-the-art method, GeneSplicer. Our predictive models show significant improvement in accuracy. Our final feature set, which includes a mix of feature types, achieves a 4.4% improvement in the 11-point average precision when compared to GeneSplicer. At the 95% sensitivity level, our method yields a 10% improvement in specificity.

Our contribution is two-fold. First, we give a general feature generation framework appropriate for any sequence data problem. Second, we provide new results for splice-site prediction that should be of great interest to the gene-finding community.

2 Related Work

Feature selection techniques have been studied extensively in text categorization[1–5]. Recently they have begun receiving more attention for applications to biological data. Liu and Wong [6] give a good introduction for filtering methods used for the prediction of translation initiation sites. Degroves et al. [7] describe a wrapper approach which uses both SVMs and Naive Bayes to select the relevant features for splice sites. Other recent work includes models based on maximum entropy [8], in which only a small neighborhood around the splice site is considered. Zhang et al. [9] propose a recursive feature elimination approach using SVM and Saeys et al. have also proposed a number of different models [10, 11]. Finally, SpliceMachine [12] is the latest addition with compelling results for splice-site prediction.

In addition, there is a significant amount of work on splice-site prediction. One of the most well-known approaches is GeneSplicer proposed by Pertea et al [13]. It combines Maximal Dependency Decomposition (MDD) [14] with second order Markov models. GeneSplicer is trained on splice-site sequences 162 nucleotides long. This splice neighborhood is larger than most other splice-site programs [15]. GeneSplicer, similar to most other programs, assumes that splice sites follow the AG/GT nucleotide-pair consensus for acceptor and donor sites respectively. It uses a rich set of features including position-specific nucleotides and upstream/downstream trinucleotides.

3 Data Description

We validate our methods on a dataset which contains $4,000$ RefSeq[3] pre-mRNA sequences. Each sequence contains a whole human gene with $1,000$ additional nucleotides before and after the annotated start and stop locations of the gene. The base alphabet is $\{a, c, g, t\}$. The sequences have a non-uniform length distribution ranging from $2,359$ nucleotides to $505,025$ nucleotides. In a pre-mRNA sequence, a human gene is a protein coding sequence which is characteristically interrupted by non-coding regions, called introns. The coding regions are called exons and the number of exons per gene in our dataset varies non-uniformly between 1 and 48. The acceptor splice site marks the start of an exon and the donor splice site marks the end of an exon. All the pre-mRNA sequences in our dataset follow the AG consensus for acceptors and GT consensus for donors.

We extract acceptor sites from these sequences. Following the GeneSplicer format, we mark the splice site and take a subsequence consisting of 80 nucleotides upstream from the site and 80 nucleotides downstream. We extract negative examples by choosing random AG-pair locations that are not acceptor sites and selecting subsequences as we do for the true acceptor sites. Our data contains 20,996 positive instances and 200,000 negative instances.

4 Feature Generation

In this section we present a number of feature types for splice-site prediction and their corresponding construction procedures. If applied naively, the construction procedures produce feature sets, which become easily intractable. To keep the number of features at manageable levels, we then propose a general purpose feature generation algorithm which integrates feature construction and selection in order to produce meaningful features.

4.1 Feature Types and Construction Procedures

The feature types that we consider capture compositional and positional properties of sequences. These apply to sequence data in general and the splice-site sequence prediction problem in particular. For each feature type we describe an incremental feature construction procedure. The feature construction starts with an initial set of features and produces the constructed set of features. Incrementally, during each iteration, it produces richer, more complex features for each level of the output feature set.

[3] http://www.ncbi.nlm.nih.gov/RefSeq/

Compositional features A k-mer is a string of k-characters.We consider the general k-mer composition of sequences for k values $2, 3, 4, 5$ and 6. Given the alphabet for DNA sequences, $\{a, c, g, t\}$, the number of distinct features is 4^k for each value of k. There is a total of $5,456$ features for the k values we consider.

Construction Method. This construction method starts with an initial set of k-mer features and extends them to a set of $(k+1)$-mers by appending the letters of the alphabet to each k-mer feature. As an example, suppose an initial set of 2-mers $F_{initial} = \{ac, cg\}$. We construct the extended set of 3-mers $F_{constructed} = \{aca, acc, acg, act, cga, cgc, cgg, cgt\}$. Incrementally, in this manner we can construct levels $4, 5$ and 6.

Region-specific compositional features Splice-site sequences characteristically have a coding region and a non-coding region. For the acceptor splice-site sequences, the region of the sequence on the left of the splice-site position (upstream) is the non-coding region, and the region of the sequence from the splice-site position to the end of sequence (downstream) is the coding region. It is expected that these regions exhibit different compositional properties. In order to capture these differences we use *region-specific k-mers*. Here we also consider k-mer features for k values $2, 3, 4, 5$ and 6. Thus the total number of features is $10,912$.

Construction Method. The construction procedure of upstream and downstream k-mer features is the same as the general k-mer method, with the addition of region indication.

Positional features Position-specific nucleotides are the most common features used for finding signals in the DNA stream data [14–16]. These features capture the correlation between different nucleotides and their relative positions. Our sequences have a length of 160 nucleotides, therefore our basic position-specific feature set contains 640 features.

In addition, we want to capture the correlations that exist between different nucleotides in different positions in the sequence. Several studies have proposed *position-specific k-mers*, but this feature captures only the correlations among nearby positions. Here we propose a *conjunctive position-specific feature*. We construct these complex features from conjunctions of basic position-specific features. The dimensionality of this kind of feature is inherently high.

Construction Method. We start with an initial conjunction of basic features and add another conjunct basic feature in an unconstrained position. Let our basic set be $F_{basic} = \{a_1, c_1, ..., g_n, t_n\}$, where, for example, a_1 denotes nucleotide *a* at the first sequence position. Now, if our initial set is $F_{initial} = \{a_1, g_1\}$, we can extend it to the level 2 set of position-specific base combinations $F_{constructed} = \{a_1 \wedge a_2, a_1 \wedge c_2, ..., g_2 \wedge t_n\}$. Incrementally, in this manner we can construct higher levels. For each iteration, if the number of conjuncts is k we have a total of $\binom{n}{k} \times 4^k$ such features for a sequence of length n.

4.2 Feature Selection

Feature selection methods reduce the set of features by keeping only the useful features for the task at hand. The problem of selecting useful features has been the focus of extensive research and many approaches have been proposed [1–3, 5, 17]. In our experiments we consider several feature selection methods to reduce the size of our feature sets, including *Information Gain (IG), Chi-Square (CHI), Mutual Information (MI)* [18] and *KL-distance (KL)* [2]. Due to space limitations, in the experiments section, we present the combination that produced the best results. We used Mutual Information to select compositional features and Information Gain to select positional features during our feature generation step.

4.3 Feature Generation Algorithm (FGA)

The traditional feature selection approaches consider a single brute force selection over a large set of all features of all different types. We emphasize a type-oriented feature selection approach. The type-oriented approach introduces the possibility of employing different feature selection models for each type set; i.e.

for a feature set whose dimensionality is not too high we may use a wrapper approach [1] in the selection step, while for a large feature type set we may use filter approaches [3]. Also, in this manner features of different types can be generated in a parallel fashion. In order to employ the information embedded in the selected features for sequence prediction, we propose the following algorithm:

- *Feature Generation.* The first stage generates the feature sets for each feature type. We start with several defined feature types. For each feature type, we tightly couple together a feature construction step and a feature selection step and, iterating through these steps, we generate richer and more complex features. We specify a feature selection method for each feature type and thus, during each iteration, eliminate a subset of features that are obtained from the construction method. These features are usually assigned a low selection score and their elimination will not affect the performance of the classification algorithm.
- *Feature Collection and Selection.* In the next stage, we collect all the generated features of different types and apply another selection step. This selection step is performed because features of a particular type may be more important for the sequence prediction. We produce a set of features originating from different feature types and different selection procedures.
- *Classification.* The last stage of our algorithm builds a classifier over the refined set of features and learns a model for the given dataset.

In addition to being computationally tractable, this feature generation approach has other advantages such as the flexibility to adapt with respect to the feature type and the possibility to incorporate the module in a generic learning algorithm.

5 Experimental Results for Splice Site Prediction

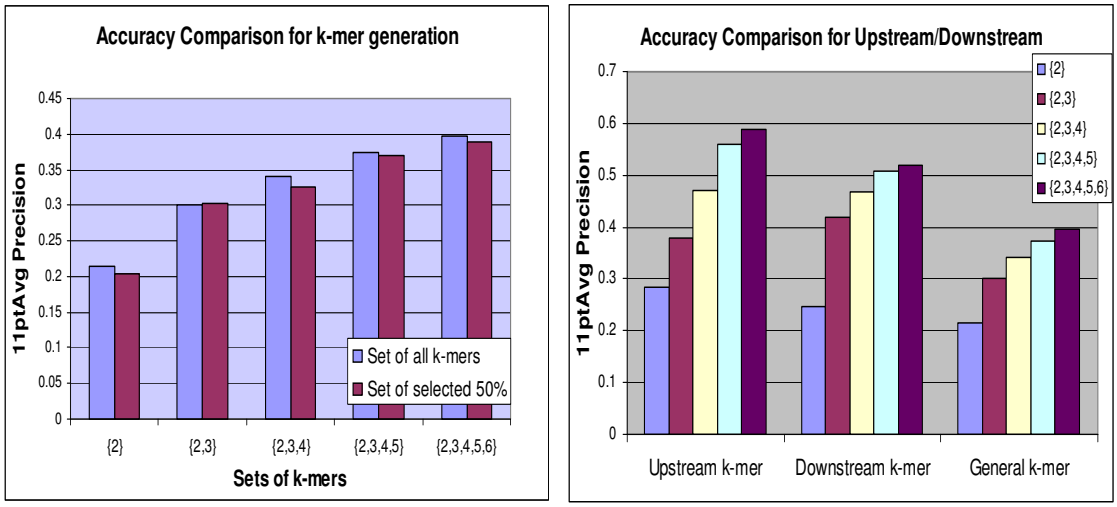
We conducted a wide range of experiments to support our claims, and here we present a summary of them. For our experiments, we considered a range of classifiers. We present results for the classifier that consistently gave the best results, called C-Modified Least Squares (CMLS) [19]. CMLS is a wide margin classifier related to Support Vector Machines (SVM), but has a smoother penalty function. This allows the calculation of gradients which can provide faster convergence.

5.1 Performance Measures

We use the *11-point average* measure[20] to evaluate the performance of our algorithm. To calculate this measure, we rank the test data in decreasing order of scores. For a threshold *t*, the test data points above the threshold are the sequences *retrieved*. Of these, those that are true positives (TP) are considered *relevant*. *Recall* is the ratio of relevant sequences retrieved to all relevant sequences (including those missed) and *precision* is the ratio of relevant sequences retrieved to all retrieved sequences. For any recall ratio, we calculate the precision at the threshold which achieves that recall ratio and compute the average precision. The 11-point average precision (11ptAVG) is the average of precisions estimated at the 11 recall values 0%, 10%, 20%, ., 100%. At each such recall value, the precision is estimated as the highest precision occurring at any rank cutoff where the recall is at least as great as that value.

The measures of *sensitivity (Se)* and *specificity (Sp)* commonly used by the computational biology community correspond respectively to the recall and precision definitions. Another performance measure commonly used for biological data is the *false positive rate(FPr)* defined as $FPr = \left(\frac{FP}{FP+TN}\right)$ where FP, and TN are the number of false positives and true negatives respectively. By varying the decision threshold of the classifier FP can be computed for all recall values. We also present results using this measure.

In all our experiments, the results reported use three-fold cross-validation.



(a) (b)

Fig. 1: (a) 11ptAVG precision results of the different collection sets of k-mers with no selection *(Sets of {2},...,{2,3,4,5,6}-mers)* and 50% of the features after using mutual information for selection *(Sets of selected 50%)* (b) Comparison between different feature type sets performances, upstream k-mers, downstream k-mers, and general k-mers shown in sets of {2},...,{2,3,4,5,6}-mers.

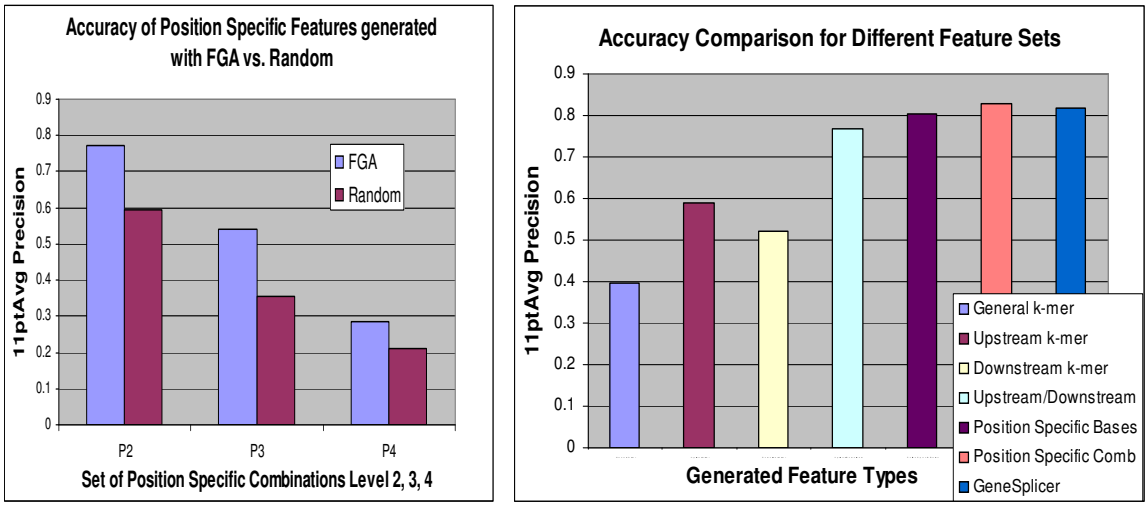
5.2 Accuracy Results of FGA

In the following experiments we present the evaluation of four different feature types, which carry positional and compositional information. As discussed in Section 4.3 initially we evaluate them separately and then identify the best group of features for each type before combining them.

Compositional features and splice-site prediction We examine each k-mer feature set independently for each value of k. We use the whole k-mer set to construct the new $(k+1)$-mer set. In our experiments, we found the *MI* selection method works best for compositional features. Figure 1(a) shows the accuracy results for the general k-mer features as we collect them through each iteration. Note that reducing the number of features in half has little effect on the overall performance. In Figure 1(b) we highlight the contribution of the region-specific k-mer features at each iteration. It is clear that k-mer features carry more information when associated with a specific region (upstream or downstream) and this is shown by the significant increase in their 11ptAVG precisions. We combine upstream and downstream k-mer features and summarize the results in Figure 2(b) along with the individual performances of each feature type. These features show an 11ptAVG precision of 77.18%, as compared to 39.84% of general k-mers.

Next, we collect the generated compositional features in the *feature collection and selection stage* of our algorithm. During this step, we pick $2,000$ compositional features of different types without affecting the performance of the classification algorithm. From this final set we observe that, in general, higher level k-mers are more informative for splice-site prediction. Furthermore, we find that the generated final k-mer feature set reveals more 5-mers and 6-mers originating from the downstream (coding) region. This is to be expected since these features can capture the compositional properties of the coding region.

Positional features and splice-site prediction Position-specific nucleotides, which constitute our basic feature set F_{basic}, give a satisfactory 11ptAvg precision, 80.34%. This is included in the graph in Figure 2(b). An initial observation of the conjunctive position-specific features reveals that, for pair-wise combinations, we have over $200,000$ unique pairs and for combinations of triples this number exceeds 40 million. Using our feature generation algorithm, we generate higher levels of this feature type, starting with the basic position-specific nucleotide features. For each conjunct level we use the construction



(a) (b)

Fig. 2: a) 11ptAvg Precision results for the position specific feature sets generated with FGA algorithm vs randomly generated features. b) Performance results of the FGA method for different feature types as well as the GeneSplicer program

method to get the next level of features. We use the *IG* selection method to select the top scoring $1,000$ features and repeat the generation to get the next level using the selected set of features as the initial set. We explore from one to four conjuncts denoted as $(P1, P2, P3, P4)$.

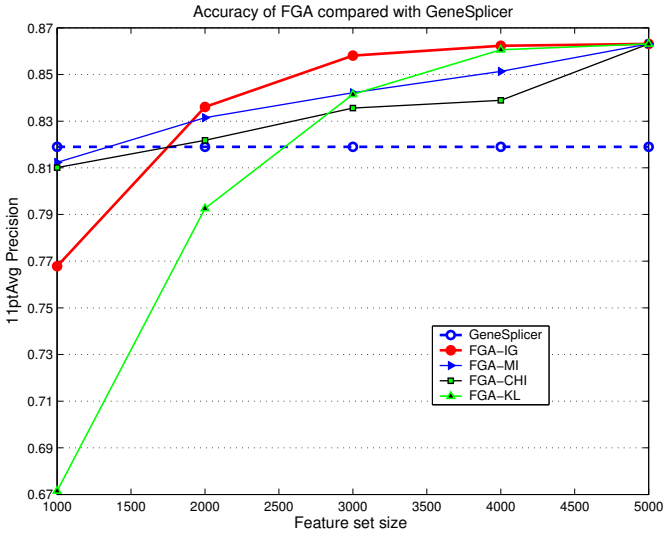
In Figure 2(a), we show the performances of the conjunctive feature sets $P2$, $P3$, and $P4$. For comparison, we introduce a baseline method, which randomly picks $1,000$ conjunctive features from each level of two, three and four conjuncts. We randomly generate 10 rounds of such feature sets from each level and we compute the average performance for the level. We compare our feature generation algorithm against this random generation baseline. As we can see from the figure, FGA outperforms random selection significantly.

In the *feature collection and selection step*, we combine the FGA generated features that carry positional information. Without any loss in 11ptAVG precision we select the top $3,000$ features of this collection. The 11ptAvg precision that this collection set gives for the acceptor splice-site prediction is 82.67% as summarized in Figure 2(b). These results clearly show that using more complex position-specific features is beneficial. In particular, we observe that pairs of position-specific bases, i.e. level 2 features, are a very important feature set that should be exploited. Interestingly, typically they are not considered by existing splice-site prediction algorithms. Figure 2(b) also shows the performance of GeneSplicer on the same dataset. We see that our positional features combination performs better than GeneSplicer.

The final collection and comparison with GeneSplicer In the following set of experiments, we show the results after we collect the features of all types that we have generated. We run our CMLS classification algorithm with a feature set of size $5,000$ containing general k-mers, upstream/downstream k-mers, position-specific nucleotides and conjunctions of position-specific features. We achieve an 11ptAVG precision performance of 86.31%. This compares quite favorably with one of the leading programs in splice-site prediction, GeneSplicer, which yields an accuracy of 81.89% on the same dataset. The precision results at all individual recall points are shown in Figure 3(a). As it can be seen from the figure, our precision results are consistently higher than those of GeneSplicer at all 11 recall points. For these experiments, in Figure 3(b), we have included the results of repeated selection for *IG, MI, CHI* and *KL* feature selection methods. Since the collection stage of our algorithm allows for several feature selection steps, we explore more aggressive feature selection options and see that smaller feature sets of even $2,000$ also

Recall	Precision(Specificity)	
(Sensitivity)	FGA	GeneSplicer
0	1	1
0.1	0.984	0.967
0.2	0.976	0.960
0.3	0.967	0.941
0.4	0.957	0.926
0.5	0.946	0.907
0.6	0.932	0.880
0.7	0.906	0.849
0.8	0.865	0.792
0.9	0.787	0.689
1	0.174	0.095
11ptAVG	0.863	0.819

(a)



(b)

Fig. 3: (a) The precision values for FGA and GeneSplicer at 11 recall points (b) 11ptAverage precision results for FGA varying the feature set size, compared to GeneSplicer

outperform GeneSplicer. Of these, we prefer the *IG* selection method since it retains the high precision of greater than 86% and in such problems of biological nature a higher specificity is very important.

In order to give further details on the difference between the performances of the two programs we present the false positive rates for various sensitivity values in Figure 4. Our feature generation algorithm, with its rich set of features, consistently performs better than GeneSplicer. Our false positive rates are favorably lower at all recall values. At a 95% sensitivity rate the FPr decreased from 6.2 to 4.3%. This is a significant reduction in false positive predictions. This can have a great impact when splice-site prediction is incorporated into a gene-finding program.

6 Conclusions

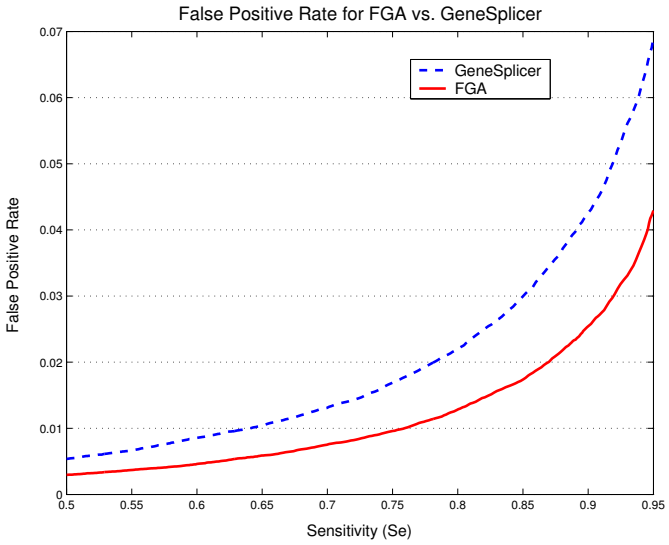
We presented a general feature generation framework, which integrates feature construction and feature selection in a flexible manner. We showed how this method could be used to build accurate sequence classifiers. We presented experimental results for the problem of splice-site prediction. We were able to search over an extremely large space of feature sets effectively, and we were able to identify the most useful set of features of each type. By using this mix of feature types, and searching over combinations of them, we were able to build a classifier which achieves an accuracy improvement of 4.4% over an existing state-of-the-art splice-site prediction algorithm. The specificity values are consistently higher for all sensitivity thresholds and the false positive rate has favorably decreased. In future work, we plan to apply our feature generation algorithm to more complex feature types and other sequence prediction tasks, such as translation start site prediction.

References

1. Kohavi, R., John, G.: The wrapper approach. In: *Feature Extraction, Construction and Selection : A Data Mining Perspective*, Liu,H.,Motoda,H.,eds. Kluwer Academic Publishers (1998)
2. Koller, D., Sahami, M.: *Toward optimal feature selection*. In: ICML. (1996) 284–292
3. Yang, Y., Pedersen, J.: *A comparative study on feature selection in text categorization*. In: ICML. (1997)

Sensitivity (Recall)	FPr (False Positive Rate)	
	FGA	GeneSplicer
0.97	0.059	0.092
0.95	0.043	0.062
0.93	0.033	0.056
0.92	0.030	0.051
0.90	0.025	0.043
0.87	0.020	0.035
0.85	0.017	0.030
0.82	0.015	0.025
0.80	0.012	0.022
0.75	0.009	0.017
0.70	0.007	0.013
0.65	0.010	0.006

(a)



(b)

Fig. 4: (a) The false positive ratio values for FGA and GeneSplicer at various sensitivity thresholds (b) The false positive rate results for FGA varying the sensitivity threshold, compared to GeneSplicer

4. Yu, L., Liu, H.: *Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution*. In: ICML. (2003)
5. Blum, A., Langley, P.: *Selection of relevant features and examples in machine learning*. Artificial Intelligence (1997)
6. Liu, H., Wong, L.: *Data mining tools for biological sequences*. Journal of Bioinformatics and Computational Biology (2003)
7. Degroeve, S., Baets, B., de Peer, Y.V., Rouze, P.: *Feature subset selection for splice site prediction*. In: ECCB. (2002) 75–83
8. Yeo, G., Burge, C.: *Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals*. In: RECOMB. (2003)
9. Zhang, X., Heller, K., Hefter, I., Leslie, C., Chasin, L.: *Sequence information for the splicing of human pre-mRNA identified by support vector machine classification*. Genome Research **13** (2003) 2637–2650
10. Saeys, Y.: *Feature selection for classification of nucleic acid sequences*. PhD thesis, Ghent U., Belgium (2004)
11. Saeys, Y., Degroeve, S., Aeyels, D., de Peer, Y.V., Rouze, P.: *Fast feature selection using a simple estimation of distribution algorithm: a case study on splice site prediction*. Bioinformatics **19** (2003) ii179–ii188
12. Degroeve, S., Saeys, Y., Baets, B.D., Rouz, P., de Peer, Y.V.: *SpliceMachine: predicting splice sites from high-dimensional local context representations*. Bioinformatics **21** (2005) 1332–1338
13. Pertea, M., Lin, X., Salzberg, S.: *GeneSplicer: a new computational method for splice site prediction*. Nucleic Acids Research **29** (2001) 1185–1190
14. Burge, C., Karlin, S.: *Prediction of complete gene structures in human genomic DNA*. Journal of Molecular Biology (1997) 78–94
15. Kim, W., Wilbur, W.: *DNA Splice Site Detection: A Comparison of Specific and General Methods*. In: AMIA. (2002)
16. Zhang, M.: *Statistical features of human exons and their flanking regions*. Human Molecular Genetics **7** (1998) 919–932
17. Brank, J., Grobelnik, M., Frayling, N.M., Mladenic, D.: *Interaction of feature selection methods and linear classification model*. In: Workshop on Text Learning. (2002)
18. Mitchell, T.: *Machine Learning*. The Mc-Graw-Hill Companies, Inc. (1997)
19. Zhang, T., Oles, F.: *Text categorization based on regularized linear classification methods*. Information Retrieval **4** (2001) 5–31
20. Witten, I., Moffat, A., Bell, T., eds.: *Managing Gigabytes*. 2 edn. Van Nostrand Reinhold (1999)