Analysis of Effect of the Position on Weighted Degree Kernel for Splice Site Prediction

Tian-Qi WANG[1, a], Yong XU[2]

[1]Bio-Computing Research Center, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, China

[2]Bio-Computing Research Center, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, China

[a]email: 601011586@qq.com

Keywords: Splice site prediction, Weighted degree kernel, Position factor, Support vector machine.

Abstract. Prediction of splice sites plays a key role in the annotation of genes. SVM with the weighted degree kernel has been proved to achieve a satisfactory performance. However, this kernel did not consider the effect of the position. In this article, we explored the relationship between the weighted degree kernel and the position of single base match. We defined a position factor to measure the effect of the position on weighted degree kernel, and selected several positions with high position factors to be key positions. Then we constructed a classification model and applied it to the Homo sapiens splice site dataset. To analyze the effect of the position of single base match, we removed the base in the key position and compared the classification accuracy with the accuracy without removing. The result shows that the position of single base match has significant influence on weighted degree kernel.

Introduction

Owing to the tremendous increase in genomic sequence data, there is an urgent demand to improve the efficiency of computational algorithms for gene annotation [1]. The accurate prediction of splice sites plays a key role in the annotation of genes in eukaryotes [2]. Most eukaryotic protein-coding genes are composed of exons and introns. In transcription introns are removed by RNA splicing so they are not coded into protein. The border between an exon and intron is referred to as a splice site. Obviously, there exists two kinds of splice sites, the splice site at the beginning of the intron is termed as a donor site, and the splice site at the end of the intron is termed as an acceptor site. Many studies have shown that most introns have an almost invariant dinucleotide GT in a donor site and AG in an acceptor site. Unfortunately, there are a large number of GT and AG dinucleotides in eukaryotic genes, but only 0.1% of them are true splice sites [3]. How to identify whether or not a GT/AG dinucleotide is a true splice site is always one of the most important and challenging tasks in bioinformatics [3, 4]. In this article, we refer to sequences with true splice sites as positives and sequences with false splice sites as negatives.

In the literature, several statistical models have been constructed for splice site prediction. The weight matrix method (WMM) [5] is the earliest and most influential one that uses the position-specific compositional biases. Subsequently, many pattern recognition algorithms, such as artificial neural network [6], hidden Markov model [7], Bayesian network [8], support vector machine (SVM) [3], etc., were successively applied to this work. In recent years, SVM and kernel method have been used frequently

for splice site prediction due to their high accuracy and capability to deal with high-dimensional and large scale datasets [9]. However, the classification accuracy of most of SVMs depends on the feature extraction significantly. Actually, for sequence analysis tasks such as splice site prediction, the string kernel is a simple but efficient method without the feature extraction. In this article, we mainly use weighted degree (WD) kernel which is a kind of string kernel and this method can compute efficiently without even extracting and enumerating all words from the sequences [10].

Many studies have shown that the base in the position close to the splice site is conserved, so it can be applied to splice site prediction. However, the WD kernel only takes the length information of the base match at the responding position into account but ignores the position information of the match. In this article, we show that the position of single base's match has significant influence on performance of WD kernel, and this can improve the performance of the WD kernel.

Methods

Weighted Degree Kernel

The so-called weighted degree (WD) kernel is a kind of string kernel [11]. It is defined as

$$k(x_1, x_2) = \sum_{\omega=1}^{d} w_\omega \sum_{i=1}^{N-d} I\left(u_{\omega,i}(x_1) = u_{\omega,i}(x_2)\right). \tag{1}$$

The main idea of the WD kernel is to count the matches between two sequences $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ between the words $u_{\omega,i}(\boldsymbol{x_1})$ and $u_{\omega,i}(\boldsymbol{x_2})$ where $u_{\omega,i}(\boldsymbol{x}) = x_i x_{i+1} \cdots x_{i+\omega-1}$ for all i and $1 \le \omega \le d$ [10]. All matching substrings are rewarded with a score depending on the length of the substring. Figure 1 shows the idea appropriately. Given two sequences $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ of equal length, the kernel consists of a weighted sum to which each match in the sequences makes a contribution w_B depending on its length B, where longer matches contribute more significantly.

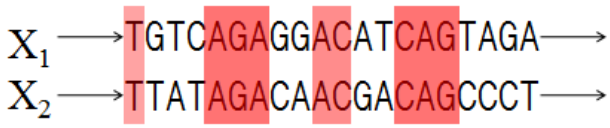

Figure 1. The idea of WD kernel

Position Factor and Key Position

From the definition of the WD kernel, we can see that the main idea of the WD kernel is to count the co-occurrence of K-mers at the corresponding position in the two sequences [12]. However, the WD kernel only considers the length information of the base match at the responding position, but many studies have shown that the base near the splice site in a certain position is highly conserved. So it will be an interesting question whether the position of the single base match has effect on the performance of the SVM classifier with the WD kernel.

Due to the fact that the base in certain position near the true splice site is conserved, it's obvious that a position, in which the base distribution of positives and negatives is

different, is more likely to be a "key position". We think that the "key positions" have greater effect on the performance of the WD kernel.

The main idea of selecting "key position" is described in Figure 2. Since the WD kernel counts the match, the "key position" that we select should be the one in which the base from positives or negatives is more conserved than in other positions. For example in Figure 2, the bases from positives in a "key position" are all "G" and the bases from negatives in the "key position" are all "A".

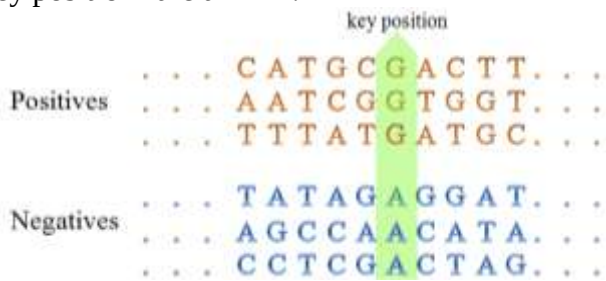

Figure 2. The main idea of selecting key positions

However, it's obvious that this is just an ideal case. So in order to measure this feature for each position i in positives, inspired by the position-weighted matrix (PWM) [13], we define d_i^+ as

$$d_i^+ = \sum_{b \in B} \left(p_{bi}^+ - \overline{p} \right)^2, \tag{2}$$

where $B = \{A, G, C, T\}$, p_{bi}^+ is probability of each base in positives and $\overline{p}$ is the average value of p_{bi}^+. Similarly, we also define d_i^- for each position i in negatives. Obviously, the greater d_i^+ and d_i^- are, the more likely the position is a key position.

However, only d_i^+ and d_i^- are not enough. If in a position the base distribution is very similar even the same between positives and negatives, such as dinucleotide GT in a donor site, d_i^+ and d_i^- cannot select the real "key position" for the WD kernel. So we define $d_i^{+,-}$ to make our method more discriminative for positives and negatives. Let v_1 and v_2 be the vectors composed of probability of each base in positives and negatives respectively. And the $d_i^{+,-}$ is defined as the Euclidean distance between v_1 and v_2. The greater $d_i^{+,-}$ is, the more likely the position is a key position.

For each position i we can get the definition of the position factor

$$f_i = \lg\left(d_i^+ \times d_i^- \times d_i^{+,-} + 1 \right). \tag{3}$$

In this article, we use this value to measure the effect of each position on WD kernel and then select the key position.

SVM

SVM is one of the most important machine learning algorithm based on statistical learning theory, which is widely-used in many fields. Based on structural risk minimization instead of empirical risk minimization, SVM can solve the problems of small-sample, non-linearity, over-fitting, dimension disaster, local minimum point, etc.,

and also has strong generalization ability [14]. The scikit-learn [15] is a practical machine learning tool-box in Python, and this study used its SVM classifier with custom kernel so as to take advantage of the WD kernel as needed.

Model Evaluation

Recall, precision and Matthew's correlation coefficients (MCC) are for determining the performance of a classification model and are defined as follows:

$$recall = \frac{tp}{tp + fn}, \tag{4}$$

$$precision = \frac{tp}{tp + fp}, \tag{5}$$

$$MCC = \frac{tp \times tn - fp \times fn}{\sqrt{(tp + fp)(tp + fn)(tn + fp)(tn + fn)}}, \tag{6}$$

where tp, fp, tn, and fn represent the number of true positives, false positives, true negatives and false negatives, respectively.

Results and Discussions

In this section we discuss experimental results that we obtained with our method of finding key positions for acceptor and donor splice sites. And then we evaluate the effect of the potential key position on WD kernel.

Dataset

To construct a reliable experiment, we used the publicly available Homo sapiens splice site dataset (HS3D) [16] as the model dataset, which was derived from human genes. The dataset contains 2796 confirmed true donor splice sites, 271,937 pseudo-donor sites, 2880 confirmed true acceptor sites, and 329,374 pseudo-acceptor sites. The redundant information has already been removed. Each splice site sequence has the length of 140 bp. For donor splice sites, the GT dinucleotide is conserved in positions 71 and 72 of the sequences, and for acceptor splice sites, AG is conserved in positions 69 and 70 of the sequences. In this article, we will use the whole dataset to calculate the position factor of each position firstly. Then we will construct a 1:1 dataset randomly and use this dataset to evaluate the key position's effect on WD kernel.

Select the Key Position

In this step, we will use the whole dataset to calculate the position factor. First we calculate each base's probability in each position in positives and negatives respectively, then we get two 140*4 position-base matrix, which is also called "PPM" [17]. Using PPM we can get the position factor for acceptor and donor easily, and the result is shown in Figure 3A and 3B.

From Figure 3A and 3B, we can see that for acceptor splice site the position factors of the position from 57 to 66, 68 and 71 of the sequences are much higher than those of other positions. And for donor splice site the position factors of the position 70 and from 73 to 75 of the sequences are much higher than those of other positions. So we consider these positions as the candidate key positions, and they may contribute a lot to the performance of the WD kernel.

The Key Position's Effect on WD Kernel

In the previous step, we have selected several candidate key positions. Now we will evaluate the effect of these candidate key positions on the performance of the WD kernel.

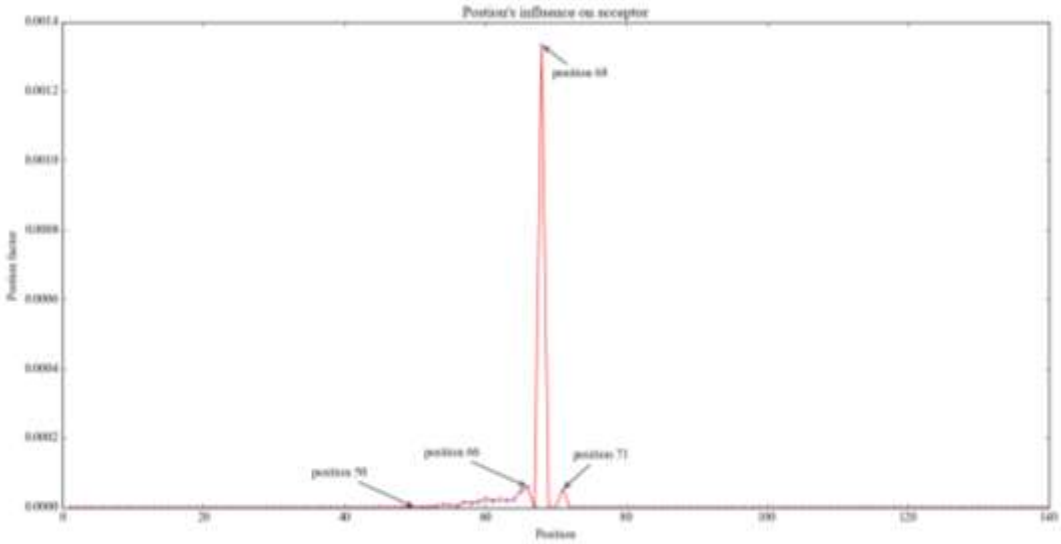

Figure 3A. The position factor for acceptor

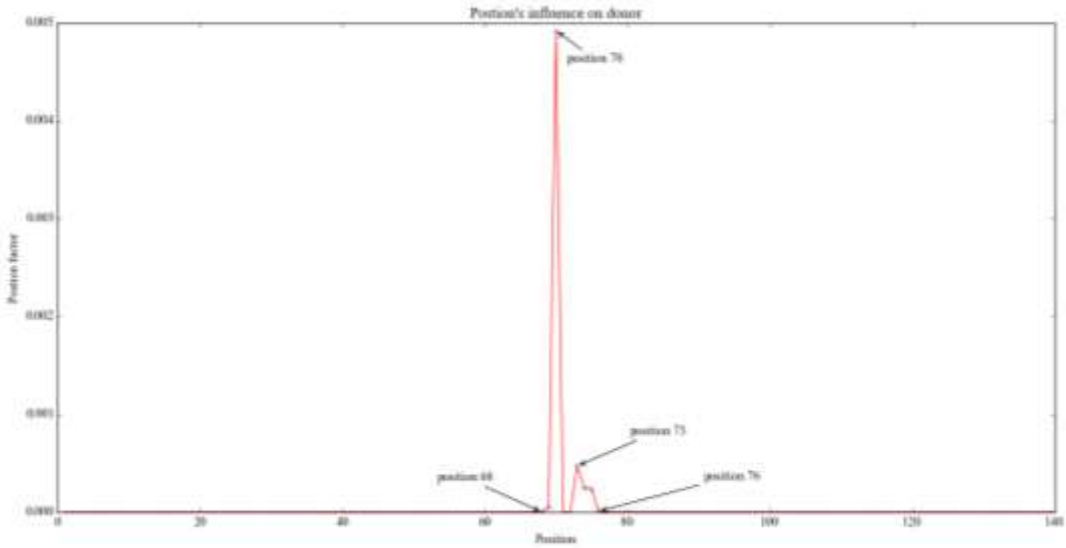

Figure 3B. The Position Factor for Donor

We selected all of the true splice sites and randomly selected the same number of pseudo-sites (2796 donor sites and 2880 acceptor sites) to construct the training set. In this case, the ratio between the number of true splice sites and that of pseudo-splice sites is 1:1. Based on the constructed 1:1 dataset, we used the SVM classifier with the WD kernel to carry out 10-fold cross-validation, and parameter d of the WD kernel was set to 12.

To show the effect of the candidate key position clearly, we first apply the information of all positions to the WD kernel. Then we remove the base in the target position for each sequence and then compare the performance after removing the base in the candidate key position to the original performance. By such comparison, the

significance of each key position on WD kernel can be available. At the same time, we also select some non key positions to do the contrary experiment.

For acceptor, we select to remove the key position 60, 66, 68, 71 as well as non-key-position 50, 67, 69, 72 for contrast experiment. For donor, we select to remove the key position 70, 73, 74, 75 as well as non-key-position 67, 68, 71, 76 for contrast experiment. The result is shown in Table 1 and Table 2.

We can see that for both acceptor and donor when we remove the base in the candidate key position, the value of recall, precision and MCC all decreased in different degree. The higher the position factor the position has, the more three kinds of value decreased. So we can get the conclusion that the position factor is a satisfactory measurement of the position's significance on WD kernel and the key positions have significance on the performance of the WD kernel.

Table 1 The result of acceptor (10-fold cross-validation)

	position	recall	precision	MCC
remove key position	60	94.97	90.53	0.8510
	66	94.82	89.87	0.8424
	68	92.27	88.96	0.8129
	71	94.78	90.01	0.8436
keep all positions	-	95.49	90.78	0.8586
remove non key position	50	95.32	90.67	0.8558
	67	95.30	90.86	0.8577
	69	95.42	90.74	0.8575
	72	95.20	90.60	0.8540

Table 2 The result of donor (10-fold cross-validation)

	position	recall	precision	MCC
remove key position	70	93.13	90.91	0.8381
	73	93.84	90.49	0.8402
	74	95.77	90.88	0.8625
	75	94.73	90.30	0.8462
keep all positions	-	96.67	93.01	0.8946
remove non key position	67	96.63	93.04	0.8946
	68	96.64	92.69	0.8908
	71	96.63	93.07	0.8948
	76	96.33	91.96	0.8801

Conclusions

For identification of splice sites, many studies reveal that the position information is useful, but string kernel method such as the WD kernel method did not take it into account. In this paper, we present a method that measures significance of the position. And using this method we find several so-called "key positions" which contribute more to the performance of the WD kernel. Based on this work, we will attempt to improve performance of the WD kernel by adding the weight to the key position in the future.

Acknowledgments

This research was partially supported by the Shenzhen Municipal Science and Technology Innovation Council (Nos. JCYJ20140904154645958).

References

[1] Li, J. L., et al. "High-accuracy splice site prediction based on sequence component and position features." *Genet Mol Res* 11.3 (2012): 3431-3451.

[2] Baten, Abdul KMA, et al. "Biological sequence data preprocessing for classification: A case study in splice site identification." Advances in Neural Networks–ISNN 2007. Springer Berlin Heidelberg, 2007. 1221-1230.

[3] Sonnenburg, Sören, et al. "Accurate splice site prediction using support vector machines." BMC bioinformatics 8.Suppl 10 (2007): S7.

[4] Baten, Abdul KMA, Saman K. Halgamuge, and Bill CH Chang. "Fast splice site detection using information content and feature reduction." BMC bioinformatics 9.Suppl 12 (2008): S8.

[5] Staden, Rodger. "Computer methods to locate signals in nucleic acid sequences." Nucleic acids research 12.1Part2 (1984): 505-519.

[6] Reese, Martin G., et al. "Improved splice site detection in Genie." Journal of computational biology 4.3 (1997): 311-323.

[7] Majoros, William H., Mihaela Pertea, and Steven L. Salzberg. "TigrScan and GlimmerHMM: two open source ab initio eukaryotic gene-finders."Bioinformatics 20.16 (2004): 2878-2879.

[8] Cai, Deyou, et al. "Modeling splice sites with Bayes networks."Bioinformatics 16.2 (2000): 152-158.

[9] Bari, ATM Golam, M. Rokeya Reaz, and Byeong-Soo Jeong. "Effective DNA Encoding for Splice Site Prediction Using SVM." MATCH-COMMUNICATIONS IN MATHEMATICAL AND IN COMPUTER CHEMISTRY 71.1 (2014): 241-258.

[10] Rätsch, Gunnar, and Sören Sonnenburg. " Accurate Splice Site Detection for Caenorhabditis elegans." Kernel Methods in Computational Biology 277 (2004).

[11] Ben-Hur, Asa, et al. "Support vector machines and kernels for computational biology." PLoS Comput Biol 4.10 (2008): e1000173.

[12] Sonnenburg, Sören, Gunnar Rätsch, and Konrad Rieck. "Large scale learning with string kernels." Large Scale Kernel Machines (2007): 73-103.

[13] Stormo, Gary D., et al. "Use of the 'Perceptron'algorithm to distinguish translational initiation sites in E. coli." Nucleic Acids Research 10.9 (1982): 2997-3011.

[14] Vapnik, Vladimir. The nature of statistical learning theory. Springer Science & Business Media, 2013.

[15] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." The Journal of Machine Learning Research 12 (2011): 2825-2830.

[16] Pollastro, Pasquale, and Salvatore Rampone. "HS3D, a dataset of Homo Sapiens splice regions, and its extraction procedure from a major public database." International Journal of Modern Physics C 13.08 (2002): 1105-1117.

[17] Guigo, Roderic. "An Introduction to Position Specific Scoring Matrices".

http://bioinformatica.upf.edu. Retrieved 12 November 2013.