Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 23 (2013) 60 – 67

Procedia Computer Science

4th International Conference on Computational Systems-Biology and Bioinformatics, CSBio2013

Recognition of promoters in DNA sequences using weightily averaged one-dependence estimators

Zaw Zaw Htike[a,]*, Shoon Lei Win[b]

[a]*Department of Electrical and Computer Engineering, Faculty of Engineering, IIUM, P.O. Box 10, 50728 Kuala Lumpur, Malaysia*
[b]*Department of Biotechnology Engineering, Faculty of Engineering, IIUM, P.O. Box 10, 50728 Kuala Lumpur, Malaysia*

Abstract

The completion of the human genome project in the last decade has generated a strong demand in computational analysis techniques in order to fully exploit the acquired human genome database. The human genome project generated a perplexing mass of genetic data which necessitates automatic genome annotation. There is a growing interest in the process of gene finding and gene recognition from DNA sequences. In genetics, a promoter is a segment of a DNA that marks the starting point of transcription of a particular gene. Therefore, recognizing promoters is a one step towards gene finding in DNA sequences. Promoters also play a fundamental role in many other vital cellular processes. Aberrant promoters can cause a wide range of diseases including cancers. This paper describes a state-of-the-art machine learning based approach called weightily averaged one-dependence estimators to tackle the problem of recognizing promoters in genetic sequences. To lower the computational complexity and to increase the generalization capability of the system, we employ an entropy-based feature extraction approach to select relevant nucleotides that are directly responsible for promoter recognition. We carried out experiments on a dataset extracted from the biological literature for a proof-of-concept. The proposed system has achieved an accuracy of 97.17 % in classifying promoters. The experimental results demonstrate the efficacy of our framework and encourage us to extend the framework to recognize promoter sequences in various species of higher eukaryotes.

Keywords: genetic sequence classification; promoter recognition; WAODE

* Corresponding author
E-mail address: zaw@ieee.org

doi:10.1016/j.procs.2013.10.009

1. Introduction

The human genome project, which was completed in the last decade, generated a perplexing mass of genetic data. Because manually analyzing genomic data is a tedious process, which resembles the process of looking for a needle in a haystack, there is a growing interest in the process of gene finding and gene recognition from DNA sequences using computational techniques. Virtually every cell in a normal eukaryote contains the same DNA. Gene expression regulation is a very important mechanism because it dictates which cells in an organism at a particular instant express which genes, out of thousands of genes contained in the DNA. Such regulation is vital for normal functioning of cells in an organism. A central regulatory region of the DNA in gene expression is the promoter region. A promoter is a segment of a DNA that marks the starting point of transcription of a particular gene. An enzyme called RNA polymerase II binds with a promoter to undergo transcription[1] wherein DNA is transcribed to become an RNA, which is then spliced to become an mRNA, which is in turn translated to come a protein as shown in Figure 1. Promoters also play a fundamental role in many other vital cellular processes. Aberrant promoters can cause a wide range of diseases including cancers. Therefore, recognizing promoters is one small step towards a giant leap in gene finding in DNA sequences[2-4].

There have been a number of attempts to perform promoter recognition using computational techniques. Altschul *et al*.[5] proposed a approach to measure similarity between two genetic sequences by a search algorithm called basic local alignment search tool (BLAST). To perform promoter recognition, one would need to store a sufficient number of examples of promoters and non-promoters. Given a novel DNA sequence, *k*-nearest neighbor classification would have to be performed where the unknown DNA sequence would be matched against all the examples in the database in pairs using the basic local alignment search. Thompson *et al*.[6] extended the idea to perform alignment search in multiple sequences rather than in pairs. However, the fundamental problems with these similarity-based approaches are that they require a wide range of examples to be stored in the database and that they are exceedingly computationally expensive. Many other computational strategies have also been proposed to

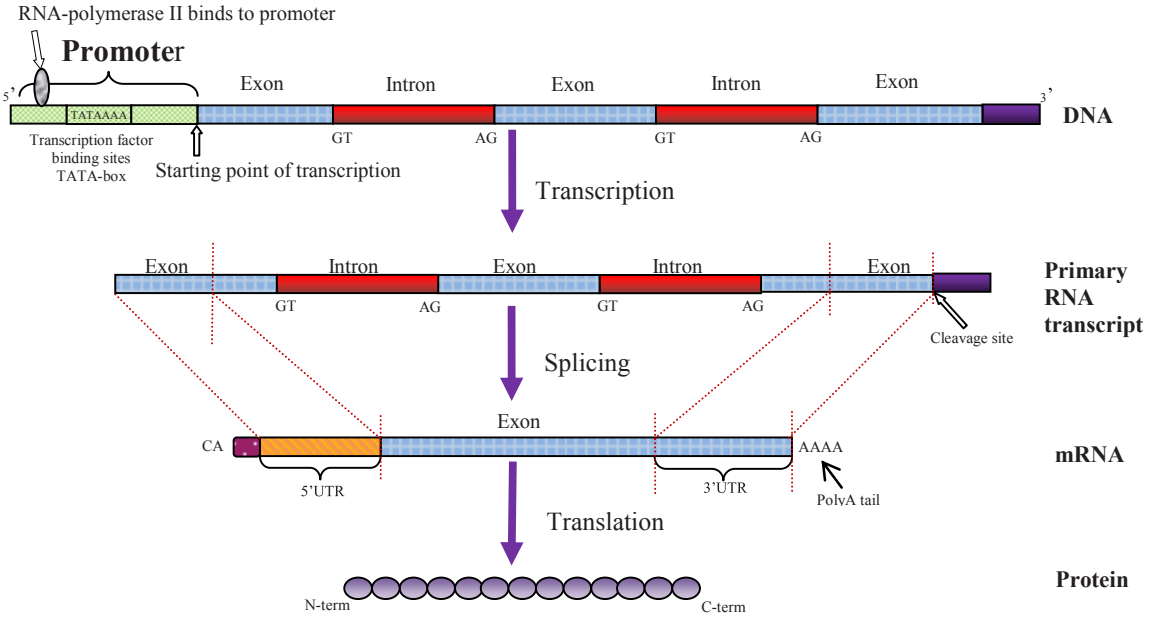

Fig. 1. Transcription in gene expression.

recognize promoters[2-4,7]. However, a vast majority of these approaches do not produce satisfactory accuracy rates in promoter recognition. Machine learning approaches have also been used to predict promoter sequences. For instance, neural networks[8-14] have been commonly used in promoter recognition. Dynamic Bayesian networks and their derivatives such as hidden Markov models (HMMs)[15] have also been used to model promoter sequences. Most of the existing approaches in the literature have high generalization errors and high computational complexity. This paper describes a state-of-the-art Bayesian approach called weightily averaged one-dependence estimators to tackle the problem of recognizing promoters from genetic sequences.

2. Promoter recognition algorithm

The goal of promoter recognition is to predict, given a sequence of nucleotides, whether or not the sequence belongs to a promoter. We propose a two-layered framework which consists of nucleotide selection and sequence classification as shown in Figure 2. The complexity of any machine learning classifier depends upon the dimensionality of the input data[16]. There is also a phenomenon known as the 'curse of dimensionality' that arises with high dimensional input data[17]. In the case of genetic sequence classification, not all the nucleotides in a genetic sequence might be responsible for discriminating between promoter and non-promoter. Therefore, we employ a nucleotide selection process to select relevant nucleotides from a given genetic sequence in an unsupervised manner. Section 2.1 describes the process of nucleotide selection. After selecting relevant nucleotides, we perform sequence classification using the weightily averaged one-dependence estimators (WAODE). Section 2.2 describes the process of classification.

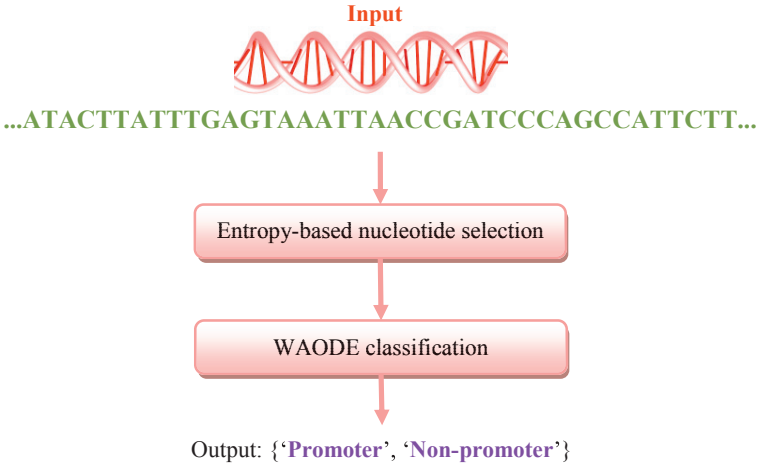

Fig. 2. High-level flow diagram of promoter recognition framework.

2.1. Entropy-based nucleotide selection

The complexity of any machine learning classifier depends upon the dimensionality of the input data~~16~~. Generally, the lower the complexity of a classifier, the more robust it is. Moreover, classifiers with low complexity have less variance, which means that they vary less depending on the particulars of a sample, including noise,

outliers, *etc*. [16] In the case of genetic sequence classification, not all the nucleotides in a genetic sequence might be responsible for discriminating promoters from non-promoters. Therefore, we need to have a nucleotide selection method that chooses a subset of relevant nucleotides that can discriminate promoters from non-promoters, while pruning the rest of the nucleotides in the input genetic sequence.

We are interested in finding the best subset of the set of nucleotides that can sufficiently discriminate promoters. Ideally, we have to choose the best subset that contains the least number of nucleotides that most contribute to the classification accuracy, while discarding the rest of the nucleotides. There are 2^n possible subsets that can arise from an n-nucleotide long genetic sequence. In essence, we have to choose the best subset out of 2^n possible subsets. Because performing an exhaustive sequential search over all possible subsets is computationally expensive, we need to employ heuristics to find a reasonably good subset that can sufficiently discriminate promoters. There are generally two common techniques: forward selection and backward selection[16]. In forward selection, we start with an empty subset and add a nucleotide (that increases the classification accuracy the most) in each iteration until any further addition of a nucleotide does not increase the classification accuracy. In backward selection, we start with the full set of nucleotides and remove a nucleotide (that increases the classification accuracy the most) in each iteration until any further removal of a nucleotide does not increase the classification accuracy. There are also other types of heuristics such as scatter search[18] and variable neighborhood search[19]. However, search-based nucleotide selection techniques do not necessarily produce the best subset of the nucleotides.

We employ a nucleotide selection process based on an information-theoretic concept of entropy. Given a set of nucleotides X and $p(x_i)$ which represents the probability of the i^{th} nucleotide, then the entropy of nucleotides, which measures the amount of 'uncertainty', is defined by:

$$H(X) = -\sum_{i=1}^{n} p(x_i)\ log\, p(x_i) \tag{1}$$

Entropy is a non-negative number. $H(X)$ is 0 when X is absolutely certain to be predicted. The conditional entropy of class label Y given the nucleotides is defined by:

$$H(Y \mid X) = \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i,\, y_j)\, ln\, \frac{p(y_j)}{p(x_i,\, y_j)} \tag{2}$$

The information gain (IG) of the nucleotides from the class label Y is defined to be:

$$IG(Y \mid X) = H(Y) - H(Y \mid X) \tag{3}$$

The gain ratio (GR) between the nucleotides and the class label Y is defined to be:

$$GR(Y \mid X) = \frac{IG(Y \mid X)}{H(Y)} \tag{4}$$

The GR of a nucleotide is a number between 0 and 1 which approximately represents the degree of 'significance' of the nucleotide in discriminating promoters. A GR of 0 roughly indicates that the corresponding individual nucleotide has no significance in promoter recognition while a GR of 1 roughly indicates that the nucleotide is significant in promoter recognition. During the training phase, the GR for each nucleotide is calculated according to (4). All the

nucleotides are then sorted by their GRs. Nucleotides whose GRs are higher than a certain threshold value are selected as discriminating nucleotides while the rest are discarded. Training needs to be carried out only once. During the recognition phase, the selected nucleotides are carried forward to classification.

2.2. Classification

Probabilistic classifiers are widely used by researchers to analyze sequences. Naive Bayes (NB) is a well-known classifier in the machine learning community owning to its simplicity, efficiency and efficacy[20-23]. NBs and their derivatives have been frequently used by researchers[24]. Unfortunately, NB is built on the strong independence assumption. NB performs fairly accurate classification. The only limitation to its classification accuracy is the accuracy of the process of estimation of the base conditional probabilities. One clear drawback is its strong independence assumption which assumes that attributes are independent of each other in a dataset. In the field of genetic sequence classification, NB assumes that nucleotides are independent of each other in a genetic sequence despite the fact that there are apparent dependencies among individual nucleotides. Semi-naive Bayesian classifiers attempt to preserve the numerous strengths of NB while reducing error by relaxing the attribute independence assumption[24]. Researchers have proposed various semi-naive techniques such as one-dependence estimators (ODEs)[25] and super parent one-dependence estimators (SPODEs)[26] to ease the attribute independence assumption. In fact, these approaches alleviate the independence assumption at the expense of computational complexity and a new set of assumptions. Webb[20] proposed a semi-naive approach called averaged one-dependence estimators (AODEs) in order to weaken the attribute independence assumption by averaging all of a constrained class of classifiers without introduction of new assumptions. The AODE has been shown to outperform other Bayesian classifiers with substantially improved computational efficiency[20]. The AODE essentially achieves very high classification accuracy by averaging several semi-naive Bayes models that have slightly weaker independence assumptions than a pure NB. The AODE algorithm is effective, efficient and offers highly accurate classification. The AODE algorithm uses the following formula for classification[24]:

$$Output = \underset{y}{\operatorname{argmax}} \left(\sum_{i:\, 1 \le i \le n \,\wedge F(x_i) \ge m} P(y, x_i) \prod_{j=1}^{n} P\left(x_j \mid y,\, x_i\right) \right) \tag{6}$$

Jiang and Zhang[27] proposed an extension of the AODE algorithm. In AODE, a special probability tree, in which each attribute is the parent of all other attributes, is built in order to augment NB. AODE essentially takes a simple average of all the nodes of this special probability tree. During classification, each node of the probability tree is treated equally. This implies that each nucleotide in a genetic sequence would be treated equally in the task of genetic sequence classification. However, in genetic sequence classification, some nucleotides in a sequence may have more influence over the others. As a result, a more natural way would be to treat each node of the probability tree differently depending on its level of 'contribution'. We approximate the level of 'contribution' of a nucleotide by measuring the mutual information between the nucleotide (X) and the output class (Y) as follows:

$$I(X;Y) = \sum_{y} \sum_{x} p(x,y) \, \log\left(\frac{p(x,y)}{p(x)\, p(y)} \right) \tag{7}$$

This improvement on AODE using variable weights for attributes is called the weightily averaged one-dependence estimators (WAODE)[27]. Note that we can also use GR or any other estimators to estimate weights for the probability tree nodes. Because the WAODE has a very weak independence assumption, it is very suitable for classification of genetic sequences. Therefore, we employ WAODE to recognize promoters in genetic sequences.

3. Experiments

We tested our proposed system using a dataset extracted from the biological literature[28]. The dataset contains 106 samples, where 50% of the samples represent promoters and the remaining 50% of the samples represent non-promoters. The positive promoter samples were taken from a compilation produced by Hawley and McClure[29]. The negative examples were derived by extracting contiguous substrings from a 1.5 kilobase sequence from a fragment from Escherichia coli bacteriophage T7 isolated with the restriction enzyme HaeIII[28]. Each sample in the dataset contains a 57 nucleotide-long DNA sequence and a label of the category ('promoter' or 'non-promoter') to which the sample represents. We performed the entropy-based nucleotide selection process on this dataset. The following 8 nucleotides were found to have the highest GRs and were consequently selected as discriminating nucleotides: 6^{th}, 15^{th}, 16^{th}, 17^{th}, 18^{th}, 20^{th}, 39^{th}, and 41^{st} (counting from left to right).

We carried out a leave-one-out cross-validation where one sample was held out as the validation data while the remaining records served as training data. The whole process was repeated multiple times such that each sample got held out exactly once as the validation data. The results were then averaged to produce an estimator to the accuracy of the proposed promoter recognition system.

Table 1 lists the summary of the leave-one-out cross-validation results. The system correctly classified a total of 103 instances out of 106 instances with an accuracy rate of 97.17% and an error rate of 2.83%. Kappa coefficient, which measures inter-rater agreement of predicted values with the true values over all the trials of the leave-one-out cross-validation, was found to be 0.9434. It means that the individual predictions are quite consistent in multiple trials and that the proposed system is robust. MAE and RMSE were found to be 0.0682 and 0.1608 respectively, which were small. RAE and RRSE were found to be significantly large. However, the RAE and RRSE metrics are not very meaningful in the task of classification. Table 2 displays the detailed results by output class. One thing interesting to note is that both true positive (TP) rate and false positive (FP) rate for promoter are lower than those for non-promoter. This implies that the system produces more negative predictions than positive predictions. This is confirmed by a lower precision score for non-promoter.

Table 1. Cross-validation results summary.

Metric	Value	
Correctly classified instances	103	(97.1698 %)
Incorrectly classified instances	3	(2.8302 %)
Kappa coefficient	0.9434	
Mean absolute error (MAE)	0.0682	
Root mean squared error (RMSE)	0.1608	
Relative absolute error (RAE)	13.5224 %	
Root relative squared error (RRSE)	31.8582 %	
Total number of instances	106	

Table 2. Detailed results by output class.

Class	TP Rate	FP Rate	Precision	Recall	F-Score	ROC Area
Promoter	0.962	0.019	0.981	0.962	0.971	0.992
Non-promoter	0.981	0.038	0.963	0.981	0.972	0.992

Table 3 compares the accuracy of the proposed system with other machine learning models and promoter recognition techniques. The proposed system, WAODE with an entropy-based nucleotide selection process produced an average error rate of 2.83%. To find out the significance of the entropy-based nucleotide selection process, we used WAODE without a nucleotide selection process to predict all the samples from the same dataset. The WAODE alone produced an error rate of 7.55%. It implies that the entropy-based nucleotide selection process does improve the overall accuracy of the system. It also implies that 8 out of 57 nucleotides are enough to discriminate promoters from non-promoters in this dataset. The error rate of the proposed promoter recognition system using the WAODE with the entropy-based nucleotide selection process seems to be lower than those of other classification systems as shown in Table 3. The results also demonstrate that WAODE outperforms AODE in genetic sequence classification. It implies that using variable weights in the probability tree does help improve classification accuracy in genetic sequence classification.

Table 3. Performance benchmark.

Technique	Avg. Error Rate (%)
WAODE with GA-based nucleotide selection	2.83
AODE[20] with GA-based nucleotide selection	3.77
KBANN[28]	3.77
SMO	6.60
WAODE without GA-based nucleotide selection	7.55
AODE[20] without GA-based nucleotide selection	10.4
RBF network	10.4
ID3[30]	11.1
O'Neill[31-32]	12.1
J48 tree	17.0

4. Conclusion

Recognizing promoters is a one step towards gene finding in DNA sequences. We have presented a machine learning based approach to recognize promoters in nucleotide sequences. NB classifiers are widely used in machine learning due to their efficiency and simplicity. However, they cannot accurately recognize nucleotide sequences because of their unrealistic assumption that forbids dependencies among individual nucleotides. We employ a state-of-the-art machine learning approach called the weightily averaged one-dependence estimators to tackle the problem of recognizing promoters in genetic sequences. Given a sequence of nucleotides, the system predicts whether the sequence belongs to a promoter. To lower the computational complexity and to increase the generalization capability of the system, we employ an entropy-based feature extraction approach to select relevant nucleotides that are directly responsible for promoter recognition. We have carried out experiments on a dataset extracted from the biological literature for a proof-of-concept. We found 8 nucleotides that were responsible for promoter recognition. This proposed system has achieved an accuracy of 97.17% in promoter recognition over this dataset. The error rate of the proposed system was found to be lower than those of other machine learning classifiers. The experimental results are quite promising. As future work, we would like to extend this framework to recognize promoter sequences in various species of higher eukaryotes. We also would like to test this framework on a wide range of datasets.

References

1. Colledge NR, Walker BR, and Ralston SH. *Davidson's Principles and Practice of Medicine*. 21st ed. 2010: Churchill Livingstone.
2. Fickett J and Hatzigeorgiou A. *Eukaryotic promoter recognition.* Genome Resolution, 1997. 7(9): p. 861-78.
3. Werner T. *Models for prediction and recognition of eukaryotic promoters.* Mammalian Genome, 1999. 10(2): p. 168-175.
4. Werner T. *The state of the art of mammalian promoter recognition.* Briefings in Bioinformatics, 2003. 4(1): p. 22-30.
5. Altschul SF et al. *Basic local alignment search tool.* Journal of Molecular Biology, 1990. 215(3): p. 403-410.
6. Thompson JD, Higgins DG, and Gibson TJ, *CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice.* Nucleic Acids Research, 1994. 22(22): p. 4673-4680.
7. Pedersen AG et al. *The biology of eukaryotic promoter prediction—a review.* Computers & Chemistry, 1999. 23(3–4): p. 191-207.
8. Conilione P and Wang D. *Neural Classification of E.coli Promoters Using Selected DNA Profiles*, in *Soft Computing as Transdisciplinary Science and Technology*, A. Abraham, et al., Editors. 2005, Springer Berlin Heidelberg. p. 51-60.
9. Demeler B and Zhou GW. *Neural network optimization for E. coli promoter prediction.* Nucleic Acids Research, 1991. 19(7): p. 1593–1599.
10. Horton PB and Kanehisa M. *An assessment of neural network and statistical approaches for prediction of E. coli promoter sites.* Nucleic Acids Research, 1992. 20(16): p. 4331-4338.
11. Mahadevan I and Ghosh I. *Analysis of E.coli promoter structures using neural networks.* Nucleic Acids Research, 1994. 22(11): p. 2158–2165.
12. Ranawana R and Palade V. *A neural network based multi-classifier system for gene identification in DNA sequences.* Neural Comput. Appl., 2005. 14(2): p. 122-131.
13. Zhang F, Kuo MD, and Brunkhors A. *E. coli promoter prediction using feed-forward neural networks*. in *International Conference of the IEEE Engineering in Medicine and Biology Society*. 2006.
14. Pedersen A and Engelbrecht J. *Investigations of Escherichia coli promoter sequences with artificial neural networks: new signals discovered upstream of the transcriptional startpoint*. in *International conference on intelligent systems for molecular biology*. 1995.
15. Pedersen AG et al. *Characterization of Prokaryotic and Eukaryotic Promoters Using Hidden Markov Models*, in *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*. 1996, AAAI Press. p. 182-191.
16. Alpaydin E. *Introduction to Machine Learning*. 2nd ed. 2010: The MIT Press.
17. Bishop CM. *Pattern Recognition and Machine Learning*. 2007: Springer.
18. García López F et al. *Solving feature subset selection problem by a Parallel Scatter Search.* European Journal of Operational Research, 2006. 169(2): p. 477-489.
19. García-Torres M et al. *Solving Feature Subset Selection Problem by a Hybrid Metaheuristic*, in *First International Workshop on Hybrid Metaheuristics*. 2004. p. 59–68.
20. Webb GI, Boughton JR, and Wang Z. *Not So Naive Bayes: Aggregating One-Dependence Estimators.* Machine Learning, 2005. 58(1): p. 5-24.
21. Hand D and Yu K. *Idiot's Bayes---Not So Stupid After All?* International Statistical Review, 2001. 69(3): p. 385-398.
22. Domingos P and Pazzani M. *On the Optimality of the Simple Bayesian Classifier under Zero-One Loss.* Mach. Learn., 1997. 29(2-3): p. 103-130.
23. Rish I. *An empirical study of the naive Bayes classifier*. in *IJCAI-01 workshop on "Empirical Methods in AI"*.
24. Zheng F and Webb GI. *Efficient lazy elimination for averaged one-dependence estimators*, in *Proceedings of the 23rd international conference on Machine learning*. 2006, ACM: Pittsburgh, Pennsylvania. p. 1113-1120.
25. Sahami M. *Learning Limited Dependence Bayesian Classifiers*. in *Second International Conference on Knowledge Discovery and Data Mining*. 1996: AAAI Press.
26. Yang Y et al. *Ensemble Selection for SuperParent-One-Dependence Estimators*, in *AI 2005: Advances in Artificial Intelligence*, S. Zhang and R. Jarvis, Editors. 2005, Springer Berlin Heidelberg. p. 102-112.
27. Jiang L and Zhang H. *Weightily Averaged One-Dependence Estimators*, in *PRICAI 2006: Trends in Artificial Intelligence*, Q. Yang and G. Webb, Editors. 2006, Springer Berlin Heidelberg. p. 970-974.
28. Towell GG, Shavlik JW, and Noordewier MO. *Refinement of approximate domain theories by knowledge-based neural networks*, in *Proceedings of the eighth National conference on Artificial intelligence - Volume 2*. 1990, AAAI Press: Boston, Massachusetts. p. 861-866.
29. Hawley DK and McClure WR. *Compilation and analysis of Escherichia coli promoter DNA sequences.* Nucleic Acids Research, 1983. 11(8): p. 2237-55.
30. Quinlan JR. *Induction of Decision Trees.* Machine Learning, 1986. 1(1): p. 81-106.
31. O'Neill MC. *Escherichia coli promoters. I. Consensus as it relates to spacing class, specificity, repeat substructure, and three-dimensional organization.* Journal of Biological Chemistry, 1989. 264(10): p. 5522-30.
32. O'Neill M and Chiafari F. *Escherichia coli promoters. II. A spacing class-dependent promoter search protocol.* Journal of Biological Chemistry, 1989. 264(10): p. 5531-4.