SCGPred: A Score-based Method for Gene Structure Prediction by Combining Multiple Sources of Evidence

Xiao Li[1]*, Qingan Ren[2], Yang Weng[2], Haoyang Cai[1], Yunmin Zhu[2], and Yizheng Zhang[1]*

[1] *College of Life Sciences, Sichuan University, Chengdu 610064, China;* [2] *College of Mathematics, Sichuan University, Chengdu 610064, China.*

Predicting protein-coding genes still remains a significant challenge. Although a variety of computational programs that use commonly machine learning methods have emerged, the accuracy of predictions remains a low level when implementing in large genomic sequences. Moreover, computational gene finding in newly sequenced genomes is especially a difficult task due to the absence of a training set of abundant validated genes. Here we present a new gene-finding program, SCGPred, to improve the accuracy of prediction by combining multiple sources of evidence. SCGPred can perform both supervised method in previously well-studied genomes and unsupervised one in novel genomes. By testing with datasets composed of large DNA sequences from human and a novel genome of *Ustilago maydi*, SCGPred gains a significant improvement in comparison to the popular *ab initio* gene predictors. We also demonstrate that SCGPred can significantly improve prediction in novel genomes by combining several foreign gene finders with similarity alignments, which is superior to other unsupervised methods. Therefore, SCGPred can serve as an alternative gene-finding tool for newly sequenced eukaryotic genomes. The program is freely available at http://bio.scu.edu.cn/SCGPred/.

Key words: gene finding, gene prediction, genome annotation, supervised method, unsupervised method, combiner method

Introduction

With the development of technologies and the efforts of genome sequencing centers all over the world, genomes of over 3,000 organisms have been sequenced or are ongoing (http://www.genomesonline.org/gold.cgi). The first stage for interpreting the genomic data is to list the protein-coding genes and determine the exact exon-intron structure for every gene. This task still remains a significant challenge, especially for eukaryotes in which coding exons are usually separated by introns of vary length (*1*). In general, there are two sorts of fundamental approaches for gene finding, that is, similarity-based methods and computational methods (*2*, *3*). In similarity-based methods, by aligning to known protein/EST (expressed sequence tag) databases or the genomes of close species, the homology matches can provide strong evidence for supporting the present of genes in a query genome. Such methods are commonly used to annotate model organisms, such as human, and to produce a "gold standard" for a reference of protein-coding genes. However, some genes that express in low level and in special tissues are missed to be annotated due to the absence of similarity evidence. In this case, computational gene finding is carried out to address the problem, and has made much progress in the last few years in terms of both methods and prediction accuracy measure. However, computational gene finding often produces predictions with a number of false positives, especially when implementing in large DNA sequences in eukaryotic genomes (*1*). Moreover, different computational gene predictors produce different and even conflict results on a same sequence, thereby confusing the users.

Accordingly, the recent trend is to combine homologous information with computational methods, which has been proven by experiments that the accuracy of prediction could be improved. The combinational approaches can be divided into two categories, strict and loose. The strict approach is to combine homologous information as condition probability into a

***Corresponding authors.**
E-mail: lix@scu.edu.cn;
yizzhang@scu.edu.cn

probabilistic model, which usually is a hidden Markov model. Currently, many computational gene finders have an extensive version to combine homologous information into their respective computing algorithm for improving predictions. The analogous programs include GenomeScan (*4*), Twinscan (*5*), SGP2 (*6*), Fgenesh+ (*7*) and AUGUSTUS+ (*8*), which are extended from GENSCAN (*9*) (the first two), GeneID (*10*), Fgenesh (*7*) and AUGUSTUS (*11*), respectively. Another group of highly integrative approach is to directly combine the predictions of several programs and/or the results of similarity searching. In contrast to the strict one, the approach is loose and capable of combining multiple and even arbitrary evidence types. The combined evidence is large and discrete in terms of type and number, including not only *ab initio* predictions, but also the results from similarity searching and other information such as predictions of splice sites. The method is exemplified by EUGENE (*12*), EGPred (*13*), Combiner (*14*) and its descendent JIGSAW (*15*).

Computational identification of protein-coding genes in novel eukaryotic genomes is especially a difficult task. Conventional statistical *ab initio* and combiner methods described above commonly use supervised machine learning methods that require a large training set of validated genes for estimating gene model parameters. For a novel genome, however, neither an abundant cDNA/EST database nor a close genome is available, thus limiting most of the methods to be employed. An alternative approach is to employ a foreign gene finder; however, Korf (*16*) demonstrated that it can produce a highly inaccurate result, even though using model parameters from a neat phylogenetic neighbor. As a result, unsupervised methods are applied as a better solution for finding genes in novel genomes, such as SNAP (*16*) and GeneMark.HMM-ES (*17*).

In this paper, we introduce a new approach, implemented in the program SCGPred (Score-based Combinational Gene Predictor), to combine multiple evidence generated from a diverse set of sources. SCGPred can perform both supervised method in well-studied genomes and unsupervised one in novel genomes. The key components of SCGPred are to deal with different and even conflict types of evidence from heterogeneous sources by a scoring system, and combine them into frame-consistent gene models using dynamic programming. We tested the performance of the supervised and unsupervised SCGPred methods on large genomic sequences from a well-studied genome (human) and a novel genome (*Ustilago maydis*) by combining four *ab initio* gene finders (GENSCAN, GeneID, Fgenesh and AUGUSTUS) with sequence alignments to protein and cDNA/EST databases. The results showed that SCGPred achieved a significant improvement (∼16%) in specificity without lossing sensitivity compared with the best of single programs, and maintained a good balance between sensitivity and specificity. Moreover, we demonstrated that SCGPred was superior to other unsupervised methods when applying in novel genomes.

Methods

There are obvious differences between evidence generated by two categories of gene prediction methods. Computational gene-finding programs, also termed as *ab initio* gene finders, use numerous mathematics models to detect signal patterns like splice sites or to distinguish content statistics between coding and non-coding regions. The evidence obtained by such method commonly refers to the actual boundaries of exons, although there are a number of false positives and false negatives. The similarity-based searching programs, such as BLASTX and Sim4, can provide strong evidence for supporting exon/intron locations in the query genomic sequence by alignments with protein and cDNA/EST databases, respectively. Intuitively, a high scoring pair (HSP) with low P-value from alignments strongly suggests the existence of a coding exon. However, these programs do not accurately delineate exon boundaries. For instance, due to the frame-shift, an HSP from alignments with protein database usually misses 1 or 2 nucleotides on boundaries compared with the actual exon. Similarity with cDNA/ESTs can often be misleading, as cDNA/EST databases are polluted by non-coding sequences, such as pre-mRNAs containing introns and untranslated regions. In addition, the similarity-based methods fail to identify exon types, because the introns of eukaryotic genes have large length if not further analyzed.

We made a collection of evidence generated from the two methods, and classified them into two groups, information evidence and validation evidence, based upon their quality, information and validation. The information evidence is the four types (initial, internal, terminal and single) of exons generated from *ab initio* gene finders. Although some gene finders predict other gene components, such as promoters and polyadenylation tracts, we only considered exon evidence in the study. Moreover, we only collected the evidence with scores because we used the scores to combine them into a framework by a probability method. Evidence from alignments with protein and EST databases was defined as validation evidence, which helps to rescore information evidence by constructing a score system. Given validation evidence, our goal was to combine information evidence into frame-consistent gene models by dynamic programming.

Construction of gene models by dynamic programming

Referring to the boundaries of four types of information evidence, we defined eight types of signals: initial-start, initial-stop, internal-start, internal-stop, terminal-start, terminal-stop, single-start and single-stop. Considering genes encoded on two strands of genomic sequence, signals are extended further into sixteen types (for example, initial-start-plus and initial-start-minus). Let L be a genomic sequence, and X is a collection of positions located on L corresponding to the signals. For a position $i \in X$, a tuple $e = [p, t, s]$ denotes the features, where p is the position on L, t is the type of signal and s is the score of signal. Because some positions have conflict evidence supported by different information sources, the tuple is extended into $e = [p, (t_1, s_1), (t_2, s_2) \ldots]$. For instance, the position x_5 in Figure 1 can be defined as $e(x_5) = [p_{x_5}$, (internal-start-plus, 0.65), (terminal-start-plus, 0.45)]. The gene prediction problem is to find a best gene model (a highest scoring gene in this case) that is assembled from the set of positions in the given X. We imposed a dynamic programming algorithm to address the problem. Let G be the set of all genes ending in each position, the score of the best gene (g) ending in a given position i for signal type t can be obtained recursively as follows:

$$S(g_{i,t}) = max\{S(g_{j,t'})\} + h(s_t) \qquad (1)$$

where h is a scoring system and $0 < j < i$. Instead of the original score, we assigned a probability score to the signal type t of position i by using a scoring system (described in details in the next subsections). For frame consistency, we computed a separate score for each of the three reading "frames" of signal type t. The frames for different signal types are defined as: the signal types of exon-starting have three phases (initial- and single-starting signals have only a phase 0), and those of exon-stopping have three reminders (terminal- and single-stopping signals have only a reminder 0). For the valid gene model, the signal type t' of position j, which can be linked to the upstream of the signal type t of position i, must be consistent with the following criteria:

1. Biological meaning. For example, the signal of initial-start-plus can only be linked back to a previous terminal-stop-plus, single-stop-plus, and initial-start-minus or single-start-minus signal type.

2. Length constraint. For different types of gene components, different length constraints are used by being calculated from the training set. For instance, if the interval of $L(j, i)$ is an intron, the length of $L(j, i)$ must be <50 bp.

3. Frame consistency. The frame of signal j must be consistent with that of signal i. For example, if the signal type of i is internal-stop-plus and that of j is internal-start-plus, the reminder $(p_i - p_j - 1)$ mod 3 of signal j is only considered when computing score of the frame 1 of signal i.

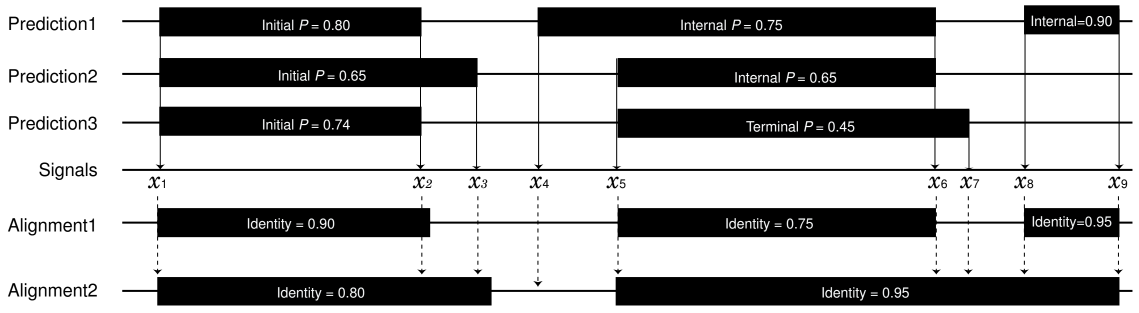

Fig. 1 Schematic illustration of SCGPred combining three *ab initio* gene predictions with results of two sequence alignments. All predicted exons have the same orientation of transcription (from left to right) and are encoded on the plus strand of a given genomic sequence. Text in each black rectangle denotes the evidence type and the corresponding probabilistic score.

The genomic sequence L is entirely scanned with the orientation from left to right ($5'$ to $3'$ in the DNA sequence) by using the dynamic programming algorithm. Each of the highest scoring genes ending in each position for each signal type is obtained and stored in G. The gene that has the best total score in G is then selected as the final gene model. If not considering the running time of evidence sources, the computational complexity is $O(N^2)$, where N is the number of signals.

Our method for constructing gene models is based upon the assumption that the highest scoring gene assembled by signals should be the best gene model. The assumption is reasonable to an actual case only when the scores of signals are probabilistic ones that are capable of measuring quantitatively the likelihood that the given signal is correct. Unfortunately, the original scores from *ab initio* gene finders are not applicable. Moreover, there are different signal types from conflict evidence in a given position, and there are different numbers of evidence for different signal types. Therefore, we constructed a scoring system to re-assign a probability to each signal. The scoring system is the first one that transforms the original scores to probabilistic ones for information evidence.

Transformation to probabilistic score

Most *ab initio* gene finders develop a scoring scheme for exon prediction, but many of them only report meaningless scores referring to the predicted exons. Although some gene finders, such as GENSCAN, give a probabilistic score to every predicted exon, the score does not respond to the likelihood correctly and is not reliable, especially when implementing in large DNA sequences (*1*). Here we applied the local polynomial regression method, a nonparametric regression model, to transform the raw scores to probabilistic ones.

Given an exon predicted by a special gene-finding program, we hope to establish the relationship between the raw score (denoted as predictor X) and the likelihood (response Y) that the exon is correct by means of a regression analysis (for example, a function m). However, the regression function m is unknown and unspecified in a simple parametric regression model. The local polynomial regression is based on the assumption that locally, near any point x, m is approximated well by a member of a simple class of parametric functions. The basic idea of the local polynomial regression consists in performing local fitting of polynomial functions by weighed least squares. More details about the theory of local polynomial were described in the literature (*18–20*). In our case, the polynomial used is of first order, and the regression method is also called as local linear regression (*21*).

We divided evenly the score scope $[x_{min}, x_{max}]$ into n small intervals, and defined the average rate of accuracy as y for every small interval. For estimating m at the score x_0 that in this case represents the score of a predicted exon, we selected a bandwidth parameter h and a kernel function K. Let X and W be the two matrix:

$$X = \begin{pmatrix} 1 & (z_1 - x_0) \\ \vdots & \vdots \\ 1 & (z_n - x_0) \end{pmatrix} \text{ and } W = diag(\frac{K(\frac{(z_i - x_0)}{h})}{h})$$

where $z_i = (x_{i-1} + x_i)/2$ and $i = 1, 2, \cdots, n$. Defining two vectors $y = (Y_1, Y_2, \ldots, Y_n)^T$ and $\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1)$, the solution of the local weighed least squares regression problem is:

$$\hat{\beta} = (X^T W X)^{-1} X^T W y \tag{2}$$

Then, the transformed score of x_0 is $\hat{\beta}_0$, namely $\hat{m}(x_0) = \hat{\beta}_0$.

We used the Epanechnikov function as the kernel function:

$$f(x) = \begin{cases} \frac{3}{4}(1 - x^2) & |x| \leq 1 \\ 0 & |x| > 1 \end{cases} \tag{3}$$

but note that a variety of other kernel functions could be used. An adequate selection of the bandwidth parameter h is crucial for a good transformation. We took $h = c^j h_{min}$ with $c = 1.1$ and $h_{min} = (x_{max} - x_{min})/2$, and got a satisfactory h on a training set by adjusting the value of j (a positive integer with $j = 1, 2, \ldots$).

The scoring system

We constructed a scoring system to assign a score to each signal in each position by combining all of the evidence. The belief degree for a given signal is supported by three aspects of information. The first one is its probabilistic scores transformed from those generated by different gene finders. Secondly, the number of gene finders that agrees with the signal among all of the gene finders can reflect its weight of belief. The third is whether the signal is at least contained in an HSP from alignments with protein or EST database.

We used the Dempster-Shafer (DS) theory of evidence (*22*) to rescore a signal in each position, which is effective especially when there are different types of signals in a given position. We defined a frame of discernment $\Theta = \{$initial, internal, terminal, single, non-coding nucleotide$\}$, and the probabilistic score of each signal as the basic belief assignment (BBA) m. All subsets of Θ form the power set denoted as 2^{Θ}, including the empty set ϕ. Let m_1 and m_2 be two BBAs defined on the same frame Θ, and be the results from two independent sources of evidence, then the joint BBA can be calculated by the following function:

$$m(A) = m_1 \oplus m_2 = \sum_{B,C \subseteq \Theta : B \cap C = A} m_1(B) \cdot m_2(C) \quad (4)$$

where A, B and C is an element or a subset on Θ. The function is known as Dempster's rule of combination. The $m(A)$ value represents a belief degree assigned to the element A. In order to map a belief measure to a probability measure, we used the pignistic transformation function (*23*) to transform the belief degree to a probability value:

$$P(\theta_i) = \sum_{\theta_i \in A \subseteq \Theta} \frac{1}{|A|} \frac{m(A)}{1 - m(\phi)} \quad (5)$$

Since an HSP cannot be determined to the type of exons, the evidence of HSP is defined as a subset of Θ that contains initial, internal, terminal and single elements. For an HSP from alignments with protein database, we extended two nucleotides respectively from the two boundaries of the HSP, so that it can contain two boundaries of the actual exon. The similarity degree (percentage of identity) is defined as the BBA of an HSP, and then the HSP is involved in the DS rule of combination.

Responding to the later two aspects of information, we used the following formula to calculate the final value of the signal t that is used to implement the dynamic programming algorithm:

$$V_t = \frac{n}{m} ds(s_t) - k_t \quad (6)$$

where the function ds is a combination of Equations 4 and 5, m is the number of sources of information evidence (*ab initio* gene finders), n is the number of gene finders that predict the signal t successfully, and k is a penalty factor for the signal type of t. For signals without validation evidence, for instance x_4 in Figure 1, we made a penalty as the signal may be most probably a false-positive. Since some ones of validation evidence (such as an HSP from the alignment with EST/cDNA database) fail to present protein-coding regions, signals with validation evidence are punished by a penalty factor as well. We used different penalty factors for different signal types, because different signal types have different measure of prediction accuracy.

Evaluation

Selection of evidence sources

Evidence that is considered for being combined is based upon the following rules: (1) it can be easily obtained for both well-studied and novel genomes; (2) the sources need to provide the scores corresponding to the evidence. We used the predictions from GENSCAN, GeneID, Fgenesh and AUGUSTUS, the four leading *ab initio* gene finders, as information evidence. All predictions were obtained by implementing with default parameters on local machines, except that Fgenesh predictions were generated on website (http://sun1.softberry.com/berry.phtml). Validation evidence for the supervised method includes the predictions from the FirstEF software (*24*) and HSPs (P<10^{-5}) from alignments with the NCBI RefSeq protein database (*25*) and the TIGR Gene Index (*26*) (including both assembled human and mouse ESTs) by using BLASTX and BLASTN programs from NCBI, respectively. For the unsupervised method, we used the UniProtKB/Swiss-Prot (*27*) and dbEST (ftp://ftp.ncbi.nih.gov/blast/db/) database alignments as validation evidence. Comparing with RefSeq, UniProtKB/Swiss-Prot does not contain computer-annotated protein sequences, thereby representing the true protein set.

Because initial exons are more poorly predicted than internal exons by prediction programs (*2*), we added the predictions of FirstEF as additional evidence for compensating the weakness. Unlike the traditional gene-finding programs as well as our method, which define initial exons beginning initiation codons, FirstEF was designed to search for the "true" first exons from transcription start sites. Therefore, the predictions of FirstEF were combined as validation evidence rather than information evidence.

Score transformation

We examined the relationship between the raw score and the prediction accuracy for all exons given by the four gene-finding programs on human chromosome 22 (chr22). The results showed that Fgenesh and GeneID cannot provide probabilistic scores. Although GENSCAN and AUGUSTUS can report a probabilistic score for each predicted exon, the raw probabilistic score fails to reflect the likelihood that the predicted exon is correct, and always appears a significantly overestimated probability (Figure 2A). Thus, the raw scores of all exons predicted by the four gene finders need to be transformed to the true probabilistic ones before combining. Further analysis showed that there are striking differences among the prediction accuracy for different exon types. For all four gene finders, internal exons have more number and higher specificity of prediction in contrast to initial, terminal and single exon types. We classified exons into three categories: internal, initial-terminal and single exons. Internal and initial-terminal exon categories were alone transformed by performing local polynomial regress method. Single exon category is not fit to perform local polynomial regress because the size of number is too small and the probability is not significant. For single exon category, we calculated the average prediction specificity on each score range and assigned the average specificity to all single exons on the range as the transformed scores.

The transformation measure by using local polynomial regress method for different score ranges of different exon types is displayed in Figure 2. The proportion of correct prediction on each score range is an average value of prediction correctness. The training set composed of exons predicted from human chr22, and those predicted from the ENCODE regions (*28*) were thought as test set. Figure 2A shows that in the case of GENSCAN and AUGUSTUS, comparing with the raw probabilistic scores of internal exons, the

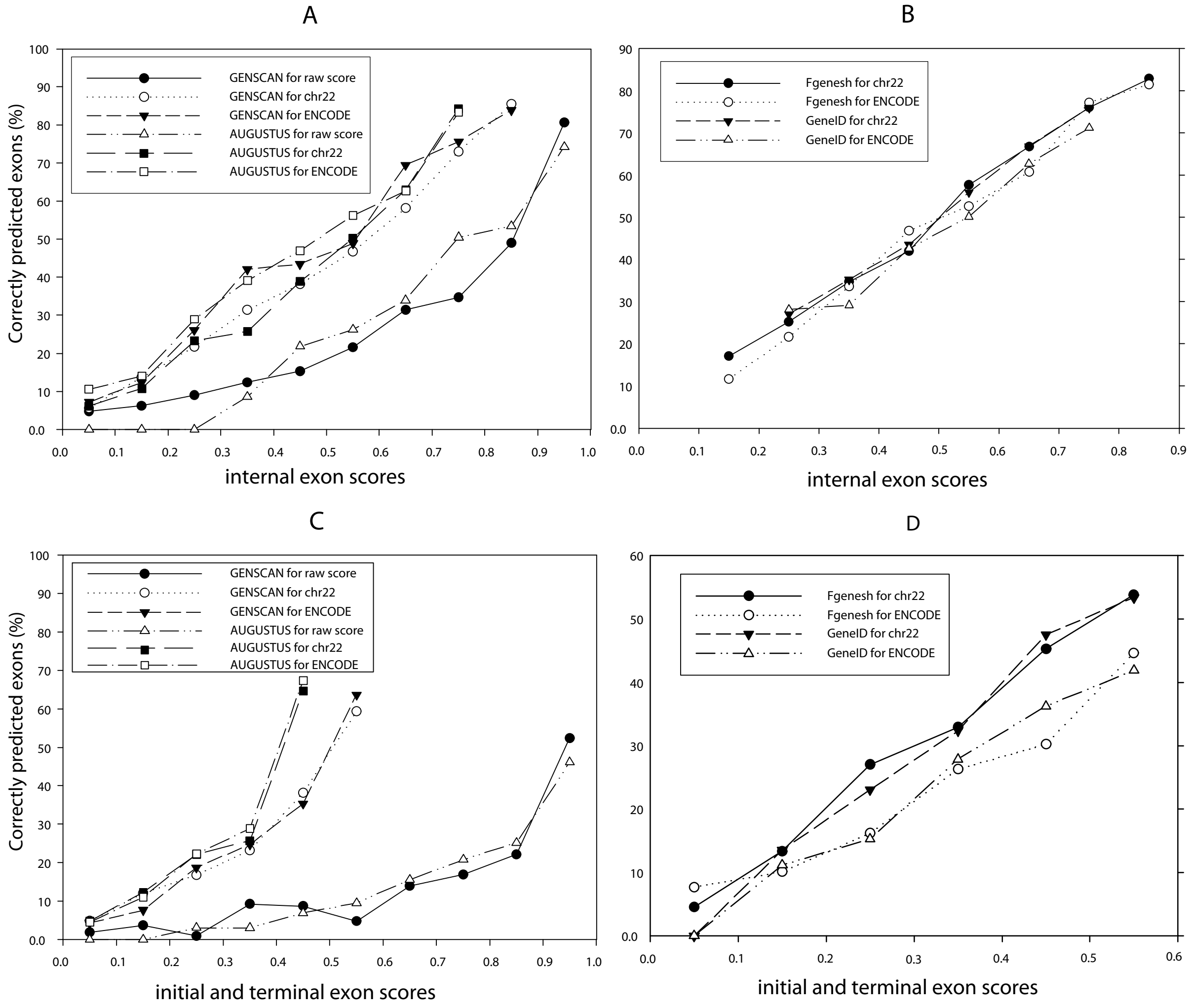

Fig. 2 The relationships between the probabilistic scores of exons predicted by the four *ab initio* gene finders (GENSCAN, AUGUSTUS, Fgenesh and GeneID) and the proportion of correctly predicted exons.

transformed ones can reflect the reliability of the predicted exons more precisely. Figure 2A and B illustrate that the transformation of internal exon scores by the four prediction programs has a good performance on both training set and test set. The transformation performance of initial and terminal exon scores is shown in Figure 2C and D, suggesting that the transformed scores of exons are also a good guide to the likelihood of correctness of prediction. However, the score transformation for initial and terminal exons is inferior to that for internal exons, especially for those predicted from GENSCAN and AUGUSTUS. The main reason is that the data size of the initial-terminal exon category on the training set is not large.

Prediction accuracy evaluations for supervised methods

We used two datasets, human chr22 and the ENCODE regions (*28*), to evaluate the accuracy of supervised SCGPred on human gene prediction. The NCBI RefSeq genes were selected as the genome annotations and considered as "gold standard" for measuring accuracy. The two datasets, including genome assembly (masked repeat elements) and the corresponding RefSeq genes, were downloaded from the UCSC Genome Browser (*29*). The genome assembly of human chr22 was divided into non-overlapped 1.5 Mb segments. All annotations were transformed to the GTF2 format (http://ardor.wustl.edu/GTF22.html) and validated by using the script "validate_gtf.pl" from Eval software package (*30*). The purpose of validating was to create a common gene ID for those with multiple transcripts from overlapping genes, which is necessary for measuring prediction accuracy by Eval program.

Human chromosome 22 has been well annotated in the last several years. Here we used the version NCBI Builds 36.1 (hg18) released on March 2006. This release of human chr22 annotated 442 genes, 560 transcripts and 3,944 exons. The ENCODE project aims to identify all functional elements on the specified 30 Mb (~1%) of human genome composed of 44 segments with at least a length of 0.5 Mb. The current annotation release of 44 ENCODE regions is hg17 using NCBI Builds 35, containing 403 genes, 528 transcripts and 3,783 exons. We removed the segment ENm004 in ENCODE regions, which is from human chr22 and overlaps the training set.

The performance of a gene predictor can be characterized generally by two terms, sensitivity and specificity, at base, exon and gene levels, respectively. Sensitivity is the fraction of positives in the test data that are predicted as positive. Specificity is the fraction of negatives in the test data that are predicted as negative. At exon level, sensitivity is the percentage of exons that are predicted correctly, and specificity is the percentage of predicted exons that are correct, so do for at base and gene levels. A notable case is that a predicted gene, which matches one of the transcripts of an annotated gene, including all exons in the transcript, is counted as a correct prediction.

We calculated the sensitivity and specificity of SCGPred prediction as well as that of the four *ab initio* gene finders combined on three levels by using the Eval program. We also estimated the performance of SCGPred combining with different gene finders by using all the validation evidence and the same penalty factors. The penalty factors used in all combinations of SCGPred were empirically derived from the training set in human chr22. The penalty factors were set at 0.7, 0.65, 0.3 and 0.6 for initial, internal, terminal and single exon types without validation evidence, and 0.1, 0.3, 0.2 and 0.25 for those with validation evidence, respectively. These penalty factors were selected as having a highest average value of exon sensitivity and specificity [(ESN+ESP)/2] on human chr22. Furthermore, we compared the results of SCGPred with those of SGP2, an extension of GeneID that combines TBLASTX alignments with mouse genome. The predictions of SGP2 were obtained from the UCSC Genome Browser.

Table 1 summarizes the test results on human chr22. The highest value at each level is indicated in bold. The results show that all the *ab initio* gene finders produce predictions with a low specificity at exon and gene levels when applied in large genomic sequences. Comparing with each of the four *ab initio* programs, SCGPred achieves an overall improvement on all levels except for base level sensitivity. SCGPred can obtain a highest performance by combining all of the gene finders. This suggests that the more evidence it combines, the higher performance SCGPred could gain. The results also show that improvements are substantially better at exon and gene levels than base level. Moreover, SCGPred is superior to SGP2 in specificity at all levels considerably, except that base and gene level sensitivity for SCGPred have slight decreases in comparison to SGP2 with 2% and 1%, respectively.

Table 1 Predication accuracy of SCGPred and other gene finders on human chromosome 22*

Method	Gene		Exon			Base	
	Sn	Sp	Sn	Sp	(Sn+Sp)/2	Sn	Sp
GENSCAN (GS)	0.09	0.05	0.71	0.40	0.56	**0.89**	0.48
GeneID (GI)	0.16	0.09	0.68	0.55	0.62	0.83	0.63
Fgenesh (FS)	0.15	0.09	0.73	0.53	0.63	0.86	0.61
AUGUSTUS (AG)	0.19	0.09	0.67	0.53	0.60	0.83	0.60
SCGPred:							
GS	0.08	0.08	0.65	0.65	0.65	0.75	0.70
GS+GI	0.15	0.14	0.69	0.68	0.69	0.78	0.71
GS+GI+FS	0.18	0.16	0.73	0.68	0.71	0.82	0.71
GS+GI+FS+AG	0.21	**0.18**	**0.74**	**0.70**	**0.72**	0.83	**0.73**
SGP2	**0.22**	0.14	**0.74**	0.60	0.67	0.85	0.69

*The highest value at each level is indicated in bold. Sn, sensitivity; Sp, specificity.

Table 2 Predication accuracy of SCGPred and other gene finders on ENCODE regions*

Method	Gene		Exon			Base	
	Sn	Sp	Sn	Sp	(Sn+Sp)/2	Sn	Sp
GENSCAN (GS)	0.10	0.04	0.67	0.38	0.52	**0.87**	0.43
GeneID (GI)	0.13	0.05	0.62	0.48	0.55	0.82	0.48
Fgenesh (FS)	0.15	0.05	**0.71**	0.43	0.57	0.87	0.44
AUGUSTUS (AG)	0.15	0.07	0.58	0.53	0.56	0.78	0.59
SCGPred:							
GS	0.08	0.06	0.61	0.64	0.63	0.71	0.61
GS+GI	0.14	0.10	0.64	0.66	0.65	0.77	0.60
GS+GI+FS	0.18	0.10	0.70	0.60	0.65	0.83	0.56
GS+GI+FS+AG	**0.20**	**0.14**	0.70	**0.69**	**0.70**	0.80	0.66
SGP2	0.14	0.10	0.70	0.61	0.66	0.85	**0.77**

The results on the ENCODE regions are given in Table 2. Specificity on three levels for all methods on the ENCODE regions are lower than that on human chr22, implying that the ENCODE regions might be annotated less completely than human chr22. Similarly to the results on human chr22, the performance of SCGPred is beyond the four *ab initio* methods. SCGPred achieves a notable accuracy of prediction on all levels, especially exon level specificity with 16% increase by comparing with the best single program (AUGUSTUS). The results in Table 2 confirm the facts that SCGPred gains a highest performance by combining all of the gene finders, and is superior to SGP2 in specificity on exon and gene levels. We compared our results indirectly with those of other combiner methods by referencing the paper (*31*), showing that SCGPred does not exceed the best methods (JIGSAW for example) when employing the supervised procedure in the ENCODE regions.

Prediction accuracy evaluations for unsupervised methods

We applied SCGPred to a novel genome of *Ustilago maydi* for creating predictions by combining the above four foreign gene finders with alignment results. *U. maydis* is a pathogenic basidiomycete fungus with a genome size of 20.5 Mb and 6,902 annotated gene models (*32*). We did not use any training set for generating predictions in the genome, therefore our method is classified into an unsupervised one. The 274 genomic scafford sequences and genome annotation files were downloaded from the Broad Institute's Fungal Genome Initiative candidate genome website (http://www.broad.mit.edu/annotation/genome/ustilago_maydis/home.html).

We created SCGPred predictions by combing the four gene finders using different parameter models. Firstly, all the four gene finders were running with human parameters, which are the most common parameters for most gene finders. Secondly, for each gene finder, parameters were selected from the nearest phylogenetic neighbors of *U. maydis* (GENSCAN still used human as parameter model due to the absence of other compatible parameters). We used directly the score transformation system that was derived from human chr22. Furthermore, we compared the result with that of the unsupervised gene finder GeneMark.HMM-ES 3.0 (*17*), which uses an iterative self-training procedure, and with that of Agene (*33*), which is a supervised gene finder training with 700 *U. maydis* annotated genes. The predictions of GeneMark.HMM-ES were obtained by running on the website (http://opal.biology.gatech.edu/GeneMark/gmseuk.cgi).

Table 3 lists the results of all programs on the novel genome. We find that employing a foreign gene finder can produce highly inaccurate results. Although the foreign gene finders can improve their predictions by using parameters from the near phylogenetic neighbors, they still remain a poor result. In contrast, SCGPred can significantly improve the prediction, especially for specificity at all levels and gains a highest improvement by combining the four foreign gene finders using the nearest phylogenetic neighbor parameters with alignment results. Comparing with the four *ab initio* gene finders, SCGPred decreases substantially the number of false positive predictions, which is crucial to reduce the risk for validating exons and genes experimentally. Moreover, SCGPred has shown consistently better performance than the unsupervised GeneMark.HMM-ES model on all levels, except that base level sensitivity is slightly lower. Agene as a supervised method produces the worst of prediction because the size of its training set is too small.

SNAP (*16*) is another program for finding genes in novel genomes by an unsupervised bootstrapping procedure. It uses a gene finder for a foreign species to create a first prediction, and then the prediction is used as virtual training set for the final gene finder. However, SNAP needs to choose a suitable foreign gene finder and requires an amount of information of genomic and gene structures for employing the bootstrapping procedure, whereas the information for most newly sequenced species including *U. maydis* is often not available. Lomsadze *et al* (*17*) compared GeneMark.HMM-ES with the supervised SNAP model, showing that GeneMark.HMM-ES had a better performance, and they inferred that it should also outperform the unsupervised SNAP model. Therefore, SCGPred has the highest prediction performance comparing with other unsupervised methods.

Parameter estimation

Penalty factors are main parameters that affect the performance of SCGPred prediction directly. We investigated the changes of the performance of SCGPred on human chr22 by using the combinations of different penalty factors for different exon types. Figure 3A displays the relationship between accuracy measure of exon level and penalty factor of internal exons without validation evidence when penalty factors of other exon types are given. From Figure 3A, a trend can be observed clearly that with the increase

Table 3 Predication accuracy of SCGPred and other gene finders on *U. maydis* genome*

Method	Parameter model	Gene		Exon			Base	
		Sn	Sp	Sn	Sp	(Sn+Sp)/2	Sn	Sp
GENSCAN	human	0.35	0.47	0.33	0.35	0.34	0.75	0.90
GeneID	human	0.27	0.26	0.25	0.23	0.24	0.75	0.90
	yeast	0.42	0.28	0.32	0.27	0.30	0.81	0.87
Fgenesh	human	0.33	0.44	0.31	0.38	0.35	0.77	0.93
	Neurospora crassa	0.40	0.48	0.39	0.43	0.41	0.77	0.94
AUGUSTUS	human	0.41	0.39	0.35	0.37	0.36	0.81	0.93
	Aedes aegypti	**0.56**	0.46	0.47	0.43	0.45	**0.85**	0.90
SCGPred	human	0.42	0.51	0.38	0.47	0.43	0.73	0.95
	phylogenetic neighbors	0.55	**0.60**	**0.47**	**0.59**	**0.53**	0.77	**0.96**
GeneMark-ES	–	0.52	0.56	0.45	0.53	0.49	0.82	0.94
Agene	*U. maydis*	0.16	0.22	0.36	0.34	0.35	0.83	0.95

*The highest value at each level is indicated in bold. Sn, sensitivity; Sp, specificity.

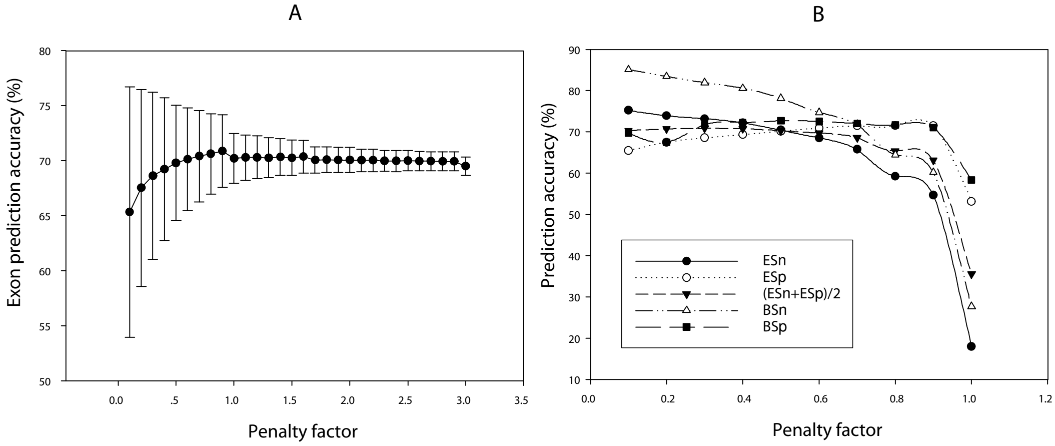

Fig. 3 Prediction accuracy of SCGPred on human chromosome 22 versus penalty factor without (**A**) and with (**B**) validation evidence. Panel A only displays at exon level, in which for every penalty factor, the top is exon sensitivity, the bottom is exon specificity, and the middle dark point represents the average value of exon sensitivity and specificity. Panel B displays accuracy results at both base and exon levels (ESn, exon sensitivity; ESp, exon specificity; BSn, base sensitivity; BSp, base specificity).

of penalty factor, sensitivity decreases and specificity increases at exon level, and the average value of sensitivity and specificity increases significantly at the beginning. Interestingly, all of them keep stable when the factor is more than 1, implying that all the exons without validation evidence have been eliminated. However, specificity of SCGPred prediction has a highest value about only 70% and increases no more in this situation, which may be due to the following two causes. Firstly, some actual genes or exons were probably missed to be annotated in the version. Secondly, false positive may be generated due to the present of pseudogenes in the validation evidence from matches with proteins or ESTs.

The suitable selection of penalty factor for internal exons with validation evidence also improves the performance of SCGPred prediction. As shown in Figure 3B, with the increase of penalty factor, sensitivity at both base and exon levels deceases, just like that without validation evidence. Specificity at both levels increases into a peak value with increasing of penalty factor, and then it deceases significantly when the penalty factor increases continuously. There is a similar appearance in the relationships between the average exon accuracy and penalty factor for internal exons with and without validation evidence. The average exon accuracy reaches a peak value with the penalty factor increasing into a certain value, which was selected by us as the optimal penalty factor and was used to compute the final results of supervised SCGPred. For the unsupervised SCGPred on novel genomes, the penalty factors should be decreased reasonably because validation evidence is not abundant.

For other exon types, the penalty factors are similar to that of internal exons for affecting sensitivity and specificity of SCGPred prediction. However, the penalty factors for other exon types affect prediction accuracy at gene level more significantly than that at base and exon levels. For a practical matter, users can tune the parameters to meet their requirements.

Conclusion

A new combiner system, SCGPred, for finding protein-coding genes has been proposed. SCGPred can apply in both well-studied and novel genomes by employing supervised and unsupervised procedures, and has a broader application scope than other combiner methods. In addition, it outperforms other unsupervised methods when applying to novel genomes. Therefore, SCGPred can serve as an alternative gene-finding tool for newly sequenced eukaryotic genomes. SCGPred is written in PERL language as a command line program, and its source code is freely available at http://bio.scu.edu.cn/SCGPred/.

Acknowledgements

The authors thank Dr. Hong Feng and Dr. Xu Song for their careful reading of the manuscript and useful suggestions. This work was partially supported by the National Natural Science Foundation of China (No. 30470984).

Authors' contributions

XL developed the method, performed data analyses and drafted the manuscript. QR, YW, HC and Y.Zhu helped with data analyses and manuscript preparation. Y.Zhang supervised the project and revised the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors have declared that no competing interests exist.

References

1. Guigó, R., *et al.* 2000. An assessment of gene prediction accuracy in large DNA sequences. *Genome Res.* 10: 1631-1642.
2. Rogic, S., *et al.* 2001. Evaluation of gene-finding programs on mammalian sequences. *Genome Res.* 11: 817-832.
3. Mathé, C., *et al.* 2002. Current methods of gene prediction, their strengths and weaknesses. *Nucleic Acids Res.* 30: 4103-4117.
4. Yeh, R.F., *et al.* 2001. Computational inference of homologous gene structures in the human genome. *Genome Res.* 11: 803-816.
5. Korf, I., *et al.* 2001. Integrating genomic homology into gene structure prediction. *Bioinformatics* 17: S140-148.
6. Parra, G., *et al.* 2003. Comparative gene prediction in human and mouse. *Genome Res.* 13: 108-117.
7. Salamov, A.A. and Solovyev, V.V. 2000. *Ab initio* gene finding in *Drosophila* genomic DNA. *Genome Res.* 10: 516-522.
8. Stanke, M., *et al.* 2006. Gene prediction in eukaryotes with a generalized hidden Markov model that uses hints from external sources. *BMC Bioinformatics* 7: 62.
9. Burge, C. and Karlin, S. 1997. Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.* 268: 78-94.
10. Parra, G., *et al.* 2000. GeneID in *Drosophila*. *Genome Res.* 10: 511-515.
11. Stanke, M. and Waack, S. 2003. Gene prediction with a hidden Markov model and a new intron submodel. *Bioinformatics* 19: ii215-225.
12. Schiex, T., *et al.* 2001. EUGENE: an eucaryotic gene finder that combines several sources of evidence. *Lect. Notes Comput. Sci.* 2066: 111-125.
13. Issac, B., *et al.* 2004. EGPred: prediction of eukaryotic genes using *ab initio* methods after combining with sequence similarity approaches. *Genome Res.* 14: 1756-1766.
14. Allen, J.E., *et al.* 2004. Computational gene prediction using multiple sources of evidence. *Genome Res.* 14: 142-148.
15. Allen, J.E. and Salzberg, S.L. 2005. JIGSAW: integration of multiple sources of evidence for gene prediction. *Bioinformatics* 21: 3596-3603.
16. Korf, I. 2004. Gene finding in novel genomes. *BMC Bioinformatics* 5: 59.
17. Lomsadze, A., *et al.* 2005. Gene identification in novel eukaryotic genomes by self-training algorithm. *Nucleic Acids Res.* 33: 6494-6506.
18. Cleveland, W. 1979. Robust locally weighted regression and smoothing scatterplots. *J. Am. Stat. Assoc.* 74: 829-836.
19. Fan, J. and Gijbels, I. 1996. *Local Polynomial Modelling and Its Applications.* Chapman & Hall, London, UK.
20. Stone, C. 1977. Consistent nonparametric regression. *Ann. Stat.* 5: 595-645.
21. Fan, J. 1993. Local linear regression smoothers and their minimax efficiencies. *Ann. Stat.* 21: 196-216.
22. Shafer, G. 1976. *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, USA.
23. Gabbay, D.M. and Smets, P. 1998. *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Volume 1. Kluwer Acedemic Publishers, Dordrecht, the Netherlands.
24. Davuluri, R.V., *et al.* 2001. Computational identification of promoters and first exons in the human genome. *Nat. Genet.* 29: 412-417.
25. Pruitt, K.D., *et al.* 2005. NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.* 33: D501-504.
26. Lee, Y., *et al.* 2005. The TIGR Gene Indices: clustering and assembling EST and known genes and integration with eukaryotic genomes. *Nucleic Acids Res.* 33: D71-74.
27. Wu, C.H., *et al.* 2006. The Universal Protein Resource (UniProt): an expanding universe of protein information. *Nucleic Acids Res.* 34: D187-191.
28. ENCODE Project Consortium. 2004. The ENCODE (ENCyclopedia Of DNA Elements) Project. *Science* 306: 636-640.
29. Hinrichs, A.S., *et al.* 2006. The UCSC Genome Browser Database: update 2006. *Nucleic Acids Res.* 34: D590-598.
30. Keibler, E. and Brent, M.R. 2003. Eval: a software package for analysis of genome annotations. *BMC Bioinformatics* 5: 50.
31. Guigó, R., *et al.* 2006. EGASP: the human ENCODE Genome Annotation Assessment Project. *Genome Biol.* 7: S2.
32. Kämper, J., *et al.* 2006. Insights from the genome of the biotrophic fungal plant pathogen *Ustilago maydis*. *Nature* 444: 97-101.
33. Munch, K. and Krogh, A. 2006. Automatic generation of gene finders for eukaryotic species. *BMC Bioinformatics* 7: 263.