BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 18 2005, pages 3596–3603
doi:10.1093/bioinformatics/bti609

Sequence analysis

JIGSAW: integration of multiple sources of evidence for gene prediction

Jonathan E. Allen[1,3,*] and Steven L. Salzberg[1,2]

[1]Center for Bioinformatics and Computational Biology and [2]Department of Computer Science, University of Maryland Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA and [3]Department of Computer Science, Johns Hopkins University, 3400 N. Charles Street, Baltimore, MD 21218, USA

Received on June 20, 2005; revised on July 28, 2005; accepted on July 29, 2005
Advance Access publication August 2, 2005

ABSTRACT

Motivation: Computational gene finding systems play an important role in finding new human genes, although no systems are yet accurate enough to predict all or even most protein-coding regions perfectly. *Ab initio* programs can be augmented by evidence such as expression data or protein sequence homology, which improves their performance. The amount of such evidence continues to grow, but computational methods continue to have difficulty predicting genes when the evidence is conflicting or incomplete. Genome annotation pipelines collect a variety of types of evidence about gene structure and synthesize the results, which can then be refined further through manual, expert curation of gene models.

Results: JIGSAW is a new gene finding system designed to automate the process of predicting gene structure from multiple sources of evidence, with results that often match the performance of human curators. JIGSAW computes the relative weight of different lines of evidence using statistics generated from a training set, and then combines the evidence using dynamic programming. Our results show that JIGSAW's performance is superior to *ab initio* gene finding methods and to other pipelines such as Ensembl. Even without evidence from alignment to known genes, JIGSAW can substantially improve gene prediction accuracy as compared with existing methods.

Availability: JIGSAW is available as an open source software package at http://cbcb.umd.edu/software/jigsaw

Contact: jeallen@umiacs.umd.edu

1 INTRODUCTION

Determining the true set of human genes has turned out to be a much harder problem than many expected; the exact number of genes has gradually decreased since the publication of the draft human genome in 2001 (International Human Genome Sequencing Consortium, 2001; Venter *et al*., 2001), but it has not stabilized. The protein-coding regions of many human genes are generally agreed upon, but even for these, the precise gene structure, comprising the boundaries of all the exons and of the coding sequence, remains less than certain. Evidential support for existing genes varies widely, from tentatively defined to experimentally confirmed. For some rarely expressed genes, the evidence is limited to a small number of expressed sequence tags (ESTs) and to the overlapping but inconsistent predictions of multiple gene finders. For some loci, evidence of expression is absent but evidence from protein sequence alignments to other species strongly suggests the presence of a gene. Many human genes have been carefully confirmed through full-length cDNA sequencing, the remapping of those cDNAs to the original chromosomes is considered the 'gold standard' for defining the true exon–intron structure. The numerous genes for which full-length cDNA sequences have not been generated pose an ongoing challenge in our efforts to produce complete and accurate predictions of all genes.

To illustrate this challenge, we consider an example from human chromosome 20, shown (Fig. 1) in a display from the UCSC genome browser, which provides an interface to a collection of programs that have been run on each human chromosome. Figure 1 shows gene predictions from several gene finding programs, plus alignments of both protein and cDNA data from Swiss-Prot (Bairoch *et al*., 2005), UniGene (Wheeler *et al*., 2003) and the TIGR Gene Index (Lee *et al*., 2005). Despite strong evidence for a gene in Figure 1, the various programs disagree on the precise gene structure. It is possible that zero, one, or multiple different predictions are correct. Currently it is left to the user to decide what evidence to use for gene structure prediction.

One track shown in Figure 1 is the Ensembl prediction (fourth row from the top), generated by the Ensembl group's automated method for integrating various forms of evidence. Ensembl, one of the leading genome annotation systems, applies a collection of rules to decide when to use the output from different prediction programs, depending on the type of evidence available for a particular gene (Curwen *et al*., 2004). The rules attempt to filter out unreliable data, leaving only high quality alignments for use in prediction. A strength of this approach is that strict criteria are established to identify high quality alignments, which is particularly useful when complete cDNA sequence can be mapped to the originating chromosome. Applying

*To whom correspondence should be addressed.

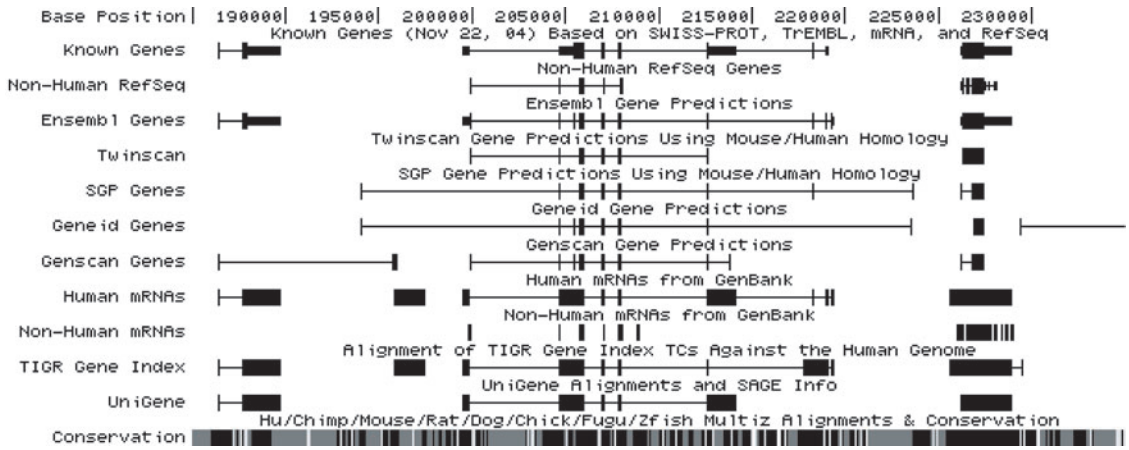

Fig. 1. Gene structure evidence from the UCSC human genome annotation database (chromosome 20). Each row shows evidence generated from a distinct source.

criteria that are too strict, however, will miss genes that are not supported by expression data (for example), even if those genes are supported by other forms of evidence. One of the goals of JIGSAW is to capture more of these genes through an automated, statistically principled method for weighing the different evidence sources.

JIGSAW uses a manually curated gene set, along with all of the alignments and predictions associated with that set, to collect statistics on the accuracy of the gene prediction evidence. The goal is to provide consistent, reproducible predictions based on sound statistical principles, even in cases of conflicting evidence about gene structure. The program is a successor to our earlier Combiner system (Allen *et al*., 2004). JIGSAW and Combiner have been used by annotators as the basis for gene calls in several recently sequenced organisms, including *Oryza sativa* (rice) (Buell *et al*., in press) and *Cryptococcus neoformans* (Loftus *et al*., 2005). This paper describes several new algorithmic developments, explains the relationship between the JIGSAW algorithm and generalized hidden Markov models (GHMMs), and evaluates JIGSAW's prediction accuracy on the human genome. Our experiments show that JIGSAW's accuracy on both exon prediction and whole-gene prediction is superior to other methods.

2 SYSTEMS AND METHODS

Computational gene finding programs are designed to model different aspects of protein coding genes, often using different statistical models for different parts of a transcript. For example, 3-periodic inhomogeneous interpolated Markov models (IMMs) have proven successful at modeling protein coding intervals (Salzberg *et al*., 1999), and decision trees (Burge and Karlin, 1997) have been used to capture dependencies between non-adjacent nucleotides near splice junctions. JIGSAW is able to take advantage of diverse models and evidence types and to combine them into frame-consistent gene predictions using dynamic programming. Here we define our dynamic programming algorithm for computing an optimally scoring parse of a genome sequence, where the parse directly corresponds to a prediction of exon–intron structure.

We represent the components of a gene with a collection of gene structure labels $Y = \{$Initial, Internal 1, Internal 2, Internal 3, Intron 1, Intron 2, Intron 3, Terminal, Single, Intergenic$\}$, with each element $y \in Y$ matching an exon, intron or intergenic region. The three distinct labels for introns and for internal exons (Internal 1, Internal 2 and Internal 3) are required to track the phase in cases where an intron interrupts a codon. Four signal types are allowed: start codons, stop codons, and splice junctions (acceptor and donor sites), which denote the beginning and ending ($5'$ and $3'$ ends) of introns. Internal exons begin one base downstream of acceptor sites and end one base upstream of donor sites. By default, the acceptor site is an AG dinucleotide, and the donor site is either a GT or GC dinucleotide. As an option, the user can replace the default set of consensus splice sites with a custom dinucleotide set. To specify which DNA strand each gene occurs on, separate labels are defined for each strand (e.g. 'Initial Plus Strand', 'Initial Minus Strand' and 'Internal 1 Plus Strand') with the Intergenic label applied to both strands. Distinct genes are not permitted to overlap even if they occur on opposite strands. Without loss of generality we define the problem for predicting genes on one strand, which can be extended to both strands using the expanded set of labels. As implemented, JIGSAW predicts genes on both strands simultaneously.

Let S be a genome sequence and $S[i, j]$ be the subsequence from position i to j (inclusive). A parse of genome sequence $S, t = (t_0, t_1, \ldots, t_n)$ consists of partitions $t_i = (b_i, e_i, y_i)$ with subsequence $S[b_i, e_i]$ assigned label y_i and t spanning the entire length of S. The parse covers each nucleotide in the sequence so that $b_{i+1} = e_i + 1$. For example, the parse for a 1000-base genome sequence containing a single-exon beginning at position 120 and ending at position 730 would be $t = [(0, 119,$ Intergenic), $(120, 730,$ Single),$(731, 999,$ Intergenic)]. The gene prediction problem is to find a parse t to maximize the joint probability of t and S: $\max_t P(t, S)$. The problem is made tractable by imposing a Markov assumption, so that label y_i in partition t_i depends only on the previous label y_{i-1} leading to $P(t, S) = \prod_{k=0}^{n} P(t_k, S)$.

Figure 2 shows a generalized hidden Markov model (GHMM) to parse genomic sequence using the Markov assumption, where states in the GHMM correspond to labels. A GHMM is defined by a set of states Q with states q and $q\prime$, and an initial state probability $P(q)$; a set of transitions from $q\prime$ to q with probability $P(q|q\prime)$; a set of probabilities $P(S[i, j]|q)$ representing the probability of generating subsequence $S[i, j]$ in state q and a set of probabilities $P_q(l)$ for the likelihood of generating a sequence of length l in state q. Using the GHMM, the joint probability for parse t and sequence S is

$$P(t, S) = \prod_k P(t_k, S) = P(S[b_0, e_0]|q_0) \cdot P(q_0) \cdot P_{q_0}(e_0 - b_0 + 1).$$

$$\prod_{k=1}^{n} P(S[b_k, e_k]|q_k) \cdot P(q_k|q_{k-1}) \cdot P_{q_k}(e_k - b_k + 1).$$

The JIGSAW dynamic programming algorithm finds the most probable parse for S of length l using an $l \times |Q|$ matrix D. Moving from left to right in the sequence, the highest scoring series of states ending in position j with

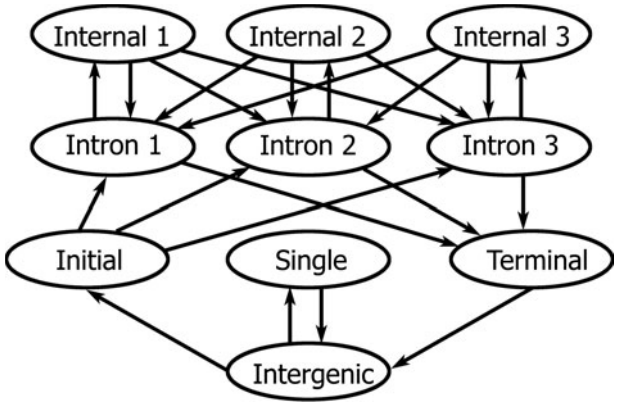

Fig. 2. Generalized hidden Markov model for predicting gene structure in genomic sequence. States represent Initial, Internal, Terminal and Single exons, respectively, plus Intron and Intergenic sequence.

state q assigned to the subsequence from i to j is stored in

$$D(j,q) = \max_{i,q'} P(S[i,j]|q) \cdot P(q|q') \cdot P_q(j-i+1) \cdot D(i,q'),$$

where $D(j,q)$ is initialized to $P(S[0,j]|q) \cdot P(q) \cdot P_q(j+1)$. The most probable parse spanning the sequence S is then found by retracing the sequence of states ending in $\max_q D(l-1,q)$. Stop codons are not allowed to span two adjacent exons within the same gene. When leaving an intron state (using the dynamic programming algorithm), the upstream and downstream exons are checked for the possible occurrence of a stop codon spanning the two adjacent exons. If a stop codon is found, the parse must end in the current intron state.

2.1 Representing gene structure evidence

Gene prediction using a set of evidence sources introduces an additional input parameter E, defined to be the gene structure evidence mapping to S. Figure 3 shows an example representation of annotation data for four sources of evidence: two gene prediction programs, GP1 and GP2 (GP2 reports a confidence score of 0.65), and two alignments to expression data with 86% and 95% identity, respectively. JIGSAW allows each evidence source to predict up to six gene features:

- start codon (sta)
- stop codon (stp)
- intron (inr)
- protein coding nucleotide (cod)
- donor (don)
- acceptor (acc)

The function

$$f_k(S,E) = \{(sta_k^0,\ldots,sta_k^{m_0}),(stp_k^0,\ldots,stp_k^{m_1}),(inr_k^0,\ldots,inr_k^{m_2}),$$
$$(cod_k^0,\ldots,cod_k^{m_3}),(don_k^0,\ldots,don_k^{m_4}),(acc_k^0,\ldots,acc_k^{m_5})\}$$

returns a set B_k of six feature vectors, one for each feature type, for each position k occurring in S. One example for each feature type is shown in Figure 3. For example, the start codon feature vector for position k_0—the evidence that a protein starts in that position—is $(1, 0.65, 0, 0)$ because GP1 and GP2 both predict the beginning of a protein here, but the sequence alignment evidence predicts a gene to start downstream at position k_1. In general, feature vector $v_{\text{type}} = v(B_k, \text{type}) = (\text{type}_k^1,\ldots,\text{type}_k^m)$ reflects what program x predicts with confidence type_k^x on nucleotide $S[k,k]$, and $g_{i,j} = g_{i,j}(E,S) = (B_i,\ldots,B_j)$ are the sets of feature vectors from position i to j. In Figure 3 the feature vector set for k_0 is

$$B_{k_0} = \{v_{\text{sta}}, v_{\text{stp}}, v_{\text{inr}}, v_{\text{cod}}, v_{\text{don}}, v_{\text{acc}}\}$$
$$= \{(1, 0.65, 0, 0), (0,0,0,0), (0,0,0,0),$$
$$(1, 0.65, 0, 0), (0,0,0,0), (0,0,0,0)\}.$$

The vector set B_{k_0} asserts that GP1 and GP2 predict a start codon at k_0, which implies a coding interval, and no other predictions overlap k_0. JIGSAW accepts any raw exon prediction score from an evidence source; however, the score should represent the program's confidence in the accuracy of its prediction. For example, the percent identity value of a transcript aligned to the target genomic sequence can be used as the confidence score, with the presumption that similar transcripts are more reliable predictors of genes than dissimilar transcripts. In cases where an evidence source does not report a confidence value (such as GP1 in Fig. 3), the entry in the feature vector is 1 or 0 to indicate the presence or absence of a prediction, respectively.

The input sequence S determines the type of prediction. For example, the cDNA alignment in Figure 3 is presumed to predict a donor site at k_3 since the alignment stops at this position and begins again at position k_4. However, the function $f_k(S,E)$ checks the sequence to ensure that a consensus splice site occurs at k_3, and k_4. Programs are assumed to predict a feature only when the feature is consistent with the sequence. The size of each feature vector - $m_0,\ldots,m_5$ can differ, so that the set of programs used to predict each type, sta, stp, inr, cod, don and acc, are assumed to be independent. Therefore, programs designed to predict only one part of the gene can be used in addition to sources that predict complete genes. Evidence for introns typically comes indirectly from gene finders and sequence alignment data. For example, in Figure 3, the evidence for an intron from k_3+1 to k_4-1 is implied by three of the four evidence sources, since each source predicts flanking exons.

In Figure 3, positions from k_4+1 to k_5-1 return the same set of feature vector values:

$$\{(0,0,0,0),(0,0,0,0),(0,0,0,0),$$
$$(1, 0.65, 0.86, 0.95),(0,0,0,0),(0,0,0,0)\}.$$

In cases like this, where $B_k = B_{k-1}$, JIGSAW compresses the two sets in one. To capture important neighboring features each B_k includes B_{k-1}, B_k and B_{k+1} (when $k=0$, B_{k-1} is defined to be a 0 valued vector).

To find the parse with the highest-score, $\max_t P(t,S,E)$, JIGSAW uses the dynamic programming matrix where $D(j,q)$ is initialized to

$$P(g_{0,j}|q, S[0,j]) \cdot P(S[0,j]|q) \cdot P(q) \cdot P_q(j+1)$$

and

$$D(j,q) = \max_{i,q'} P(g_{i,j}|q, S[i,j]) \cdot P(S[i,j]|q) \cdot P(q|q')$$
$$\cdot P_q(j-i+1) \cdot D(i,q').$$

Assuming independence, the probability of generating the feature vectors from i to j in a given state q is $P(g_{i,j}|q,S[i,j]) = \prod_{k=i}^{j} P(B_k|q,S[i,j])$, but modeling B_k poses a problem because the distribution of percent identity values from the sequence alignments cannot easily be combined with the confidence values from the gene prediction programs. Moreover, even if we assume that each of the prediction programs only generates discrete values, the number of parameters grows exponentially with respect to the number of programs in the annotation database.

2.2 Predictions conditioned on input evidence

To improve flexibility in the type and amount of evidence used, an independent conditional probability is defined for each of the six gene features, type $=$ {sta, stp, inr, cod, don, acc}. The independence assumption is justified by the fact that the collection of programs used to predict each gene feature type is assumed to be independent. In practice this is not true, since many programs are used to predict all six features (and the input sequence is the same); however, assuming independence reduces a large number of the statistical

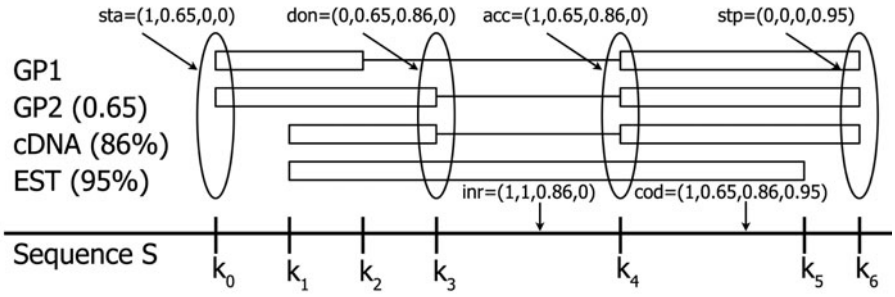

Fig. 3. Representation of four sources of gene structure evidence mapping to genome sequence S. Two gene prediction programs (GP1 and GP2), a cDNA alignment with 86% identity to S and an EST alignment with 95% identity to S. Examples of the six features, start (sta), stop (stp), coding (cod), intron (inr), donor (don) and acceptor (acc) encoded in feature vectors are shown. The predicted exon boundaries are $k_0, \ldots, k_6$.

parameters. An estimate is made for the probability of a gene feature of type occurring at position k given the feature vector for type at position k: $P(\text{type}_k|v_{\text{type}})$.

The function $h(q, k, \text{type}, S[i, j], B_k) =$

$$\begin{Bmatrix} P(\text{type}_k|v_{\text{type}}) & \text{type aligns with q} \\ 1 - P(\text{type}_k|v_{\text{type}}) & \text{otherwise} \end{Bmatrix}$$

checks to see if the gene feature, type, should occur at position k when predicting the state q to align to subsequence $S[i, j]$. As an example, when state q is *Initial*, the first nucleotide (at position i) should correspond to the beginning of a start codon and to the first coding region for a protein. In this case, $h(q, i, \text{sta}, S[i, j], B_i) = P(\text{sta}_i|v_{\text{sta}})$, $h(q, i, \text{cod}, S[i, j], B_i) = P(\text{cod}_i|v_{\text{cod}})$ and $h(q, i, \text{type}, S[i, j], B_i) = 1 - P(\text{type}_i|v_{\text{type}})$ for the four remaining feature types (stp, inr, don and acc). The probability that JIGSAW tries to compute for $P(q|g_{i,j}, S[i, j])$ when q is *Initial* is the probability of a start codon at position i given the evidence for a start codon, times the probability of a coding interval from i to j given the evidence, times the probability of a donor site at $j + 1$ given the evidence, times the probability that no conflicting features occur.

In general, the probability of state q aligning to subsequence $S[i, j]$ given all the feature vectors $g_{i,j}$ between i and j is the product of probabilities for each set of feature vectors B_k: $P(q|g_{i,j}, S[i, j]) = \prod_{k=i}^{j} \prod_{\text{type}} h(q, k, \text{type}, S[i, j], B_k)$, where *type* enumerates over all six gene features. The model makes the simplifying assumption that each feature vector set, B_k, is independent. Note that the Intergenic state is not explicitly modeled. Intergenic sequence is defined to be the absence of evidence predicting gene features in the sequence:

$$P(\text{Intergenic}|g_{i,j}, S[i, j]) = \prod_{k=i}^{j} \prod_{\text{type}} 1 - P(\text{type}_k|v_{\text{type}}).$$

A gene finder like JIGSAW that uses other gene finders as input should build on the success of existing gene finders rather than duplicating their function. Therefore, we construct a probabilistic model to compute the probability of a parse conditioned on the input evidence. [Related work in natural language processing (Sarawagi and Cohen, 2004) has demonstrated useful applications of conditional probabilities in graphical models.] The most probable parse t given S, $\max_t P(t|S, E)$ is used to make predictions. The inference algorithm for finding $\max_t P(t|S, E)$ uses the dynamic programming matrix

$$D(j, q) = \max_{i,q'} P(q|g_{i,j}, S[i, j]) \cdot P(q|q') \cdot D(i, q'),$$

where $D(j, q)$ is initialized to $P(q|g_{0,j}, S[0, j]) \cdot P(q)$.

For each scored parse, the six feature types contribute to each independent probability, in some cases predicting support for the parse and in other cases predicting support against the parse. Since each possible parse is scored using the same fixed number of independent random feature type events, the length of an exon, intron or intergenic sequence interval is dependent solely on the evaluation from the feature type models and the state transition probabilities.

2.3 Parameter estimation

Feature models are estimated for $P(\text{sta}|v_{\text{sta}})$, $P(\text{stp}|v_{\text{stp}})$, $P(\text{inr}|v_{\text{inr}})$, $P(\text{cod}|v_{\text{cod}})$, $P(\text{don}|v_{\text{don}})$ and $P(\text{acc}|v_{\text{acc}})$ through enumeration over labeled sequences in a training set. Each feature model, sta, stp, inr, cod, don and acc, is trained independently. As an example, in Figure 3, the index into f_{k_0} for start codons is $v_{\text{sta}} = (1, 0.65, 0, 0)$. The training procedure counts the number of times the evidence $(1, 0.65, 0, 0)$ occurs in the training data and the percentage of cases where $(1, 0.65, 0, 0)$ correctly predicts the start codon location. The operating assumption is the more evidence predicting a start codon with high confidence the greater chance that a start codon actually occurs. Thus, the training set should show that when all sources of evidence predict a start codon with high confidence [e.g. $v_{sta} = (1, 1, 1, 1)$] the probability of a start codon is much higher than when no evidence predicts a start codon [e.g. $v_{sta} = (0, 0, 0, 0)$].

Simple counting methods do not accurately estimate actual probability values because the sample space is theoretically infinite (in practice it is finite but extremely large). For example, $(1, 0.66, 0, 0)$ may not occur in the training set, while $(1, 0.65, 0, 0)$ happens to occur 50 times showing 80% accuracy, resulting in $P(\text{sta}|(1, 0.65, 0, 0)) = 0.8$ and leaving $P(\text{sta}|(1, 0.66, 0, 0))$ undefined. To avoid the problem of a large sample space, the observed feature vectors are first divided into two groups, accurate and inaccurate. For each feature vector v_{type} , $c(v_{\text{type}})$ is the percentage of cases in which v_{type} is observed to correctly predict *type*. v_{type} is assigned to the group accurate if $c(v_{\text{type}}) > 0.5$ and inaccurate otherwise.

A decision tree is induced to partition the feature vector space into subregions, distinguishing feature vectors in the accurate set from feature vectors in the inaccurate set. For the human genome data, JIGSAW uses the OC1 (Murthy *et al.*, 1994) decision tree system to create these trees. Each element of the feature vector is tested for 'yes' or 'no' questions to separate the accurate set from the inaccurate set. From the example in Figure 3, a trivial one-node decision tree is 'Is GP2's confidence value >0.5?', which partitions the data into two disjoint sets. These sets can be partitioned further by building a larger tree, but OC1 implements procedures to avoid partitioning the data too finely. It does this by withholding 10% of the data to determine when to stop building the tree. As the tree is built, its classification accuracy is tested on the hold-out set, and tree-building terminates when the classification accuracy drops below a threshold.

The decision tree captures in an automated way a set of rules that is similar to those used in a rule based system such as Ensembl, where percent identity cutoffs are used to make a prediction. For example, a simple prediction rule might be that a gene is valid if predicted by one gene finder and by alignment to a non-human RefSeq protein with 98% nucleotide identity. Figure 4 shows example protein coding (cod) feature vectors for non-human RefSeq

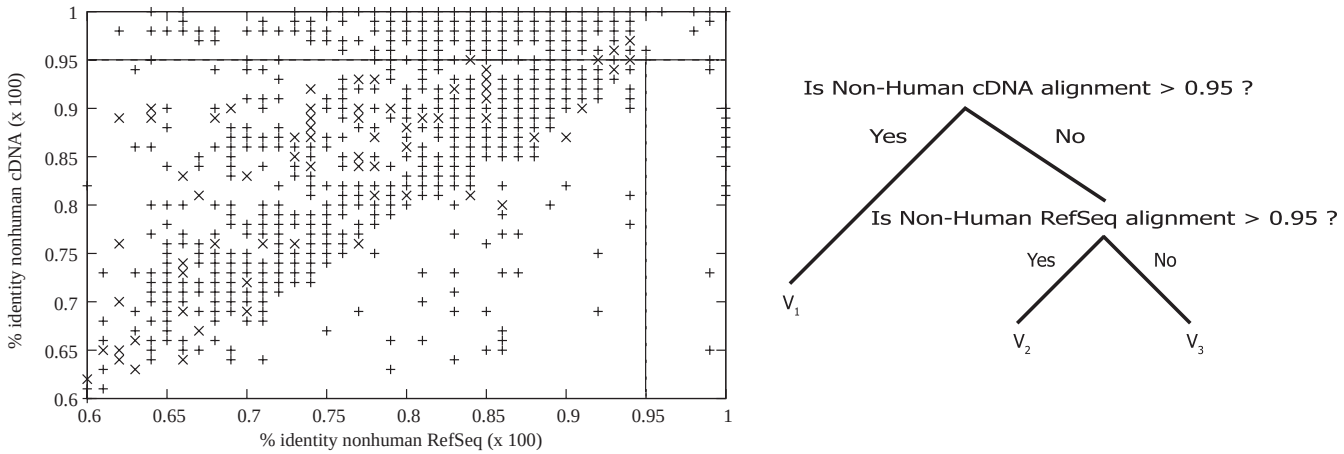

Fig. 4. The plot on the left side of the figure shows the accuracy of predictions based on alignments to non-human sequences that overlap a gene finder's predictions. Each point is a pair of alignments observed in training and their percent identity to the genomic sequence. '+' points are labeled 'accurate' and 'x' points are labeled 'inaccurate.' The two lines correspond to the non-leaf nodes in the decision tree shown on the right side of the figure.

alignments and non-human cDNA alignments, where a gene finder made an overlapping prediction. Each point in Figure 4 shows the percent identity of the respective alignments. An example is marked '+' if it is in the accurate class and 'x' if it is in the inaccurate class. The decision tree creates three partitions, which determine the cutoff values. Examples in the training set show that a threshold of 95% for either source of alignments is an accurate predictor of a protein coding region. When both alignments are <95%, the accuracy is questionable.

We can still make use of the examples below the 95% percent identity cutoff using the average probability from the individual examples. For feature vector v_{type} and decision tree β_{type}, $V = \beta_{\text{type}}(v_{\text{type}})$ is the list of the feature vectors from training, grouped into a local region. In Figure 4, evaluating the evidence vector (0.58,0.52) returns the set of examples in V_3. The probability estimate is the average accuracy of the individual examples in V_3. In general, the probability of *type* given feature vector v_{type} is $P(\text{type}|v_{\text{type}}) = \sum_{v \in V} c(v)/|V|$, where $|V|$ is the size of V.

In the earliest Combiner implementations (Allen *et al*., 2004), each evidence source was assigned an independent weight, which was shown to be effective in some cases. However, assuming independence precludes the inclusion of potentially useful sources of evidence, such as predictions from a single gene finder using two different parameter settings. The JIGSAW training procedure addresses the problem of interdependence by evaluating the accuracy of each observed evidence combination. If observed correlated sources produce similar predictions the decision tree induction algorithm can ignore a redundant source. When differences are observed, the decision tree can treat the case where correlated evidence sources overlap differently from the case where the evidence sources do not overlap.

3 RESULTS AND DISCUSSION

We used two test sets to evaluate the accuracy of JIGSAW on human gene prediction. For the first, we selected 1563 genes at random (uniformly distributed among the 24 chromosomes) from a set of 17 477 non-redundant RefSeq genes (Pruitt *et al*., 2005). We eliminated genes that are known to exhibit alternative splicing, although of course many alternative splice forms are still unknown and therefore some of the 1563 genes may still have multiple splice variants. We estimated accuracy using 3-fold cross validation, training on 2/3 of the data and testing on 1/3, and averaging results for the three experiments. For the second test, we used annotations from the Havana group (Ashurst *et al*., 2005) in the 44 ENCODE regions (The ENCODE Project Consortium, 2004), which span ∼1% of the human genome. The 44 ENCODE regions do not overlap with the 1563 genes from our first set and do include known alternatively spliced genes.

JIGSAW predictions were based on the sequence using the default consensus splice sites (GT/GC and AG) and a collection of evidence from an annotation database. Annotation data was downloaded from the UCSC genome annotation database (http://hgdownload.cse.ucsc.edu/goldenPath/hg17/database) using NCBI Build 35; examples of this evidence are shown in Figure 1.

The main evidence types are as follows:

- cDNA from human genes
- UniGene transcripts (Wheeler *et al*., 2003)
- GenBank cDNAs matching Swiss-Prot and TrEMBL proteins (Bairoch *et al*., 2005) aligned using BLAT (Kent, 2002) to the genome with at least 98% identity
- cDNA sequences from non-human species
- RefSeq genes from non-human species
- the TIGR Gene Index (Lee *et al*., 2005) (includes both assembled human and related non-human ESTs)
- *ab initio* gene finders: Genscan (Burge and Karlin, 1997), Geneid (Guigo, 1998), GeneZilla and GlimmerHMM (Majoros *et al*., 2004)
- Alignment-based gene finders: Twinscan (Flicek *et al*., 2003) and SGP (Parra *et al*., 2003) (uses mouse-human sequence conservation)
- Predicted conserved elements from phylogenetic analysis (Siepel and Haussler, 2003)

For each of the 1563 test genes JIGSAW was run on an interval that included 50 000 bases of upstream and downstream sequence. If test genes occurred in overlapping regions, the regions were merged to form a longer contiguous sequence. At a minimum, therefore, each input sequence contains over 100 000 bases. Three prediction criteria were measured: accuracy on genes, on exons and on nucleotides.

Table 1. Prediction performance on 1563 test genes for JIGSAW and for the most accurate of the evidence sources

	Gene		Exon		Nucleotide		Missed gene	Missed exon	Wrong exon
	Sens	Spec	Sens	Spec	Sens	Spec			
JIGSAW—All	58	63	86	**89**	**90**	**98**	**13**	**9**	**1**
JIGSAW	59	**66**	**87**	**89**	**90**	**98**	14	10	**1**
JIGSAW - No Ensembl	55	62	86	88	**90**	**98**	14	10	**1**
Ensembl	62	50	85	80	85	95	16	11	3
cDNA alignments	**65**	38	84	77	82	93	20	14	3

The first three rows refer to JIGSAW predictions using different combinations of evidence: JIGSAW using all input (JIGSAW—All); excluding explicit non-human expression evidence (JIGSAW); and excluding Ensembl (JIGSAW—No Ensembl). The last two rows show accuracy for Ensembl and cDNA alignments matching Swiss-Prot/TrEMBL proteins, respectively. Columns refer to sensitivity (Sens) and specificity (Spec) for genes, for exons and for coding nucleotides. The remaining columns show the percentage of completely missed genes, completely missed exons and predicted exons contained entirely in non-coding regions. The highest performance score for each column is highlighted in bold.

A gene-level prediction was counted correct only if the entire exon structure matches the test gene, from start codon to stop codon, including all intron boundaries. Non-coding exons were not included in these tests. For an exon prediction to be counted correct both the 5′ and 3′ boundaries must match the test exon exactly. A predicted protein coding nucleotide was counted correct when it matched a protein coding nucleotide in a test gene. Sensitivity is defined here as the percentage of true genes (exons) that were correctly predicted, and specificity is the percentage of the predicted genes (exons) that are correct.

On average, Ensembl predicts 1.2 isoforms per gene locus, and the alignments matching the Swiss-Prot/TrEmbl proteins generate an average of 1.7 distinct isoforms per test gene. In order not to penalize any of the programs for predicting correct alternatively spliced genes, if any one of the predicted isoforms matched the 'true' gene, the prediction was counted as correct. However, multiple identical predictions at the exon and nucleotide level were not double-counted when computing specificity.

Table 1 shows results for the 1563 test set using different combinations of evidence, along with the two single most accurate evidence sources, Ensembl and the cDNA alignments matching Swiss-Prot/TrEMBL proteins. One objective of this study was to evaluate how closely JIGSAW predictions matched the human-curated set. When JIGSAW has access to the same information as a human curator, the goal should be to report the most accurate gene predictions possible, subject to the constraints of the available evidence. Wherever JIGSAW matches the RefSeq gene, it would appear that the program is matching the ability of human curators. The first row in Table 1, JIGSAW—All, shows JIGSAW output using all available evidence, including the gene finders, cross-species conservation and expression evidence, including known proteins plus Ensembl. Interestingly, while this version yields nearly the best performance, the second row, JIGSAW, shows the results from excluding evidence from the explicit non-human gene expression sources. Excluding the explicit non-human expression evidence results in slightly more genes and exons being missed completely, but the remaining prediction accuracy measures increase.

In theory, when JIGSAW uses both Ensembl predictions and the cDNA alignments from Swiss-Prot/TrEMBL as input, it should either match or improve upon the accuracy of the those lines of evidence. The analysis is complicated by the fact that both Ensembl and the cDNA alignments predict multiple isoforms, while JIGSAW predicts (in its current implementation) only a single isoform. This shows up in the number of whole genes predicted correctly (Gene Sensitivity in Table 1), the one accuracy measure where JIGSAW does not have the best results. Predicting multiple isoforms enables Ensembl to predict 62% of the genes exactly correct, and the aligned cDNA correctly capture 65% of the genes, compared with JIGSAW's 59%. For 50% of JIGSAW predictions not matching a RefSeq gene, Ensembl or a cDNA alignment match the test gene while predicting additional isoforms. Since JIGSAW assembles a single isoform, while looking at all of the possible local gene features, the algorithm can merge the predicted isoforms into a single gene. Despite this potential limitation, JIGSAW is able to correctly detect as many or more exons completely correct as the underlying sources while picking up 5% more of the true protein coding nucleotides and completely missing less genes and exons. Moreover, the combination of specificity and sensitivity in JIGSAW is high. With 90% of protein coding nucleotides correctly detected, 66% of JIGSAW's gene predictions exactly match a RefSeq gene, showing a strong balance between detecting genes and making accurate predictions.

Row three in Table 1 (JIGSAW—No Ensembl) shows JIGSAW performance when the Ensembl predictions were excluded from its input. While JIGSAW benefits from Ensembl input, JIGSAW is able to make predictions without Ensembl and achieves comparable performance. The overall number of correctly detected genes drops slightly, but for all other measures JIGSAW's accuracy is superior.

It is important that computational gene finders be able to identify genes even when evidence from known curated human proteins is not available. Table 2 shows results for JIGSAW when we excluded the Swiss-Prot/TrEMBL protein evidence from its input. The first two rows in Table 2 show JIGSAW's results when the curated proteins were not used as evidence, but instead we used expression evidence. Using only human expression data appears to improve performance slightly in most categories, but at the cost of missing 1% more of the genes completely and 1% more of the protein coding nucleotides. Without benefit of the cDNA alignments matching the curated proteins as input, overall performance drops; however, JIGSAW still correctly detects 88–89% of the protein coding nucleotides and 41–42% of the test genes. The third row in Table 2 shows JIGSAW's performance when we excluded all expression data, using only the gene finders and sequence conservation as evidence. While performance drops still further, the results show that JIGSAW still does better than *ab initio* gene finders.

Table 2. Prediction performance on 1563 test genes, excluding the use of curated human proteins

	Gene		Exon		Nucleotide		Missed gene	Missed exon	Wrong exon
	Sens	Spec	Sens	Spec	Sens	Spec			
JIGSAW-humanEST	**42**	**45**	**84**	**82**	88	**97**	15	**10**	**1**
JIGSAW-AllEST	41	44	83	81	**89**	**97**	14	**10**	2
JIGSAW-NoEST	18	16	73	63	84	93	**12**	16	7
Twinscan	15	14	63	63	75	90	24	26	9
SGP	12	10	68	57	80	92	**12**	19	10

The first three rows refer to JIGSAW predictions using different combinations of evidence as input: JIGSAW using human expression evidence, JIGSAW using human and non-human expression data, and JIGSAW using no expression data. The last two rows show results for the two most accurate prediction systems that do not use expression data, Twinscan and SGP. (Columns are defined in Table 1.) The highest performance score for each column is highlighted in bold.

Table 3. Prediction performance on ENCODE regions

	Gene		Exon		Nucleotide		Missed genes	Missed exons	Wrong exons
	Sens	Spec	Sens	Spec	Sens	Spec			
JIGSAW	43	**72**	**88**	**93**	92	94	3	7	**1**
JIGSAW (No Ensembl)	42	**72**	87	**93**	92	**95**	3	8	**1**
Ensembl	56	55	86	88	93	89	**2**	6	**1**
cDNA alignments	**63**	48	**88**	92	**94**	91	**2**	**4**	2
JIGSAW (No curated genes)	31	54	85	89	89	93	6	9	2
JIGSAW (No expression data)	15	24	73	76	85	85	5	17	7
Twinscan	12	20	70	65	78	86	17	23	12
SGP	11	15	69	67	82	83	5	18	9

The first two rows show JIGSAW performance using human expression evidence as input. JIGSAW (No Ensembl) uses human expression evidence but excludes Ensembl input. The next two rows show the performance of Ensembl and the cDNA alignments to Swiss-Prot/TrEMBL. JIGSAW (No curated genes) uses no Ensembl and no Swiss-Prot/TrEMBL input data. The last three rows show methods not using gene expression evidence, JIGSAW (No expression data), Twinscan and SGP, respectively. (Columns are defined in Table 1.) The highest performance score for each column is highlighted in bold.

To further measure specificity in the alternative splice site prediction programs, we looked at the ENCODE regions and the Havana manual annotations for genes with start and stop codons, no in-frame stop codons and consensus splice sites. The set includes 195 loci, with 41% of the loci producing multiple transcripts, for a total of 330 isoforms. Each transcript was compared against the different programs' predictions so that all isoforms of each gene were checked. If an exon or nucleotide was correctly identified by any one of a program's predicted isoforms, we counted it as a correct prediction, but only once. (Incorrect predictions are likewise counted only once.) Results are shown in Table 3. As expected, the alternative isoform predictors (Ensembl and the cDNA alignments derived from Swiss-Prot/TrEMBL proteins) are able to capture a higher percentage of isoforms than JIGSAW. However, both programs predict many more isoforms, and thus have substantially lower specificity than JIGSAW. At the exon level, (Exon Sensitivity and Specificity in Table 3) JIGSAW performs slightly better in both sensitivity and specificity than Ensembl and the Swiss-Prot/TrEMBL cDNA based alignments, suggesting that many of the isoforms involve small changes in the exon structure. JIGSAW's balance between sensitivity and specificity remains strong, with 92% of the protein coding nucleotides correctly detected and 72% of the gene predictions exactly matching a test gene.

The ENCODE Gene Prediction Workshop (EGASP, 2005) http://genome.imim.es/gencode/workshop2005.html held in April 2005 was a meeting whose primary purpose was to assess the accuracy of computational human gene predictions in the ENCODE regions. JIGSAW predictions were submitted for comparison with dozens of other methods. Results from the workshop support our findings that JIGSAW is able to create accurate computational predictions of human genes, and in most cases outperform both *ab initio* methods and other 'combination' methods that use homology in various forms. JIGSAW's gene predictions for the ENCODE regions are freely available for download through the UCSC genome browser (http://genome.ucsc.edu/encode).

4 CONCLUSION

Cataloging the complement of human proteins remains an important yet elusive scientific milestone. Progress continues as the evidence available to support predictions increases. Computational methods need to be able to integrate this new data quickly and effectively. JIGSAW demonstrates the benefit of combining a statistical approach to evidence evaluation with a GHMM based algorithm in order to integrate multiple, often disagreeing, sources of evidence. JIGSAW is flexible with respect to types of input, requiring only that each evidence source produce a list of coordinates on a genome sequence. The program smoothly incorporates DNA and protein sequence alignment scores as well as the confidence values produced by some gene finders. As gene structure evidence continues to grow and improve, furthermore, JIGSAW's accuracy should improve as well.

Finally, JIGSAW's gene finding accuracy benefits from having free access to the rich annotation sources inside genome databases. Public access to genome sequences and annotation not only benefits biologists working on this data, but also speeds development of better computational methods.

ACKNOWLEDGEMENTS

The authors thank William H. Majoros and Mihaela Pertea for extraction of RefSeq genes and for numerous useful comments and suggestions. This work was supported in part by the National Institutes of Health under grants R01-LM06845 and R01-LM007938. Funding to pay the Open Access publication charges for this article was provided by grant R01-LM06845 to SLS from the National Institutes of Health.

Conflict of Interest: none declared.

REFERENCES

Allen,J.E., Pertea,M. and Salzberg,S.L. (2004) Computational gene prediction using multiple sources of evidence. *Genome Research*, **14**.

Ashurst,J.L. *et al*. (2005) The Vertebrate genome annotation (Vega) database. *Nucleic Acids Res.*, **33**, 459–465.

Bairoch,A. *et al*. (2005) The universal protein resource (UniProt). *Nucleic Acids Res.*, **33**, 154–159.

Buell,C.R. *et al*. (2005) Sequence, annotation, and analysis of synteny between rice chromosome 3 and diverged grass speices. *Genome Res., in press*.

Burge,C. and Karlin,S. (1997) Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.*, **268**, 78–84.

Curwen,V. *et al*. (2004) The Ensembl automatic gene annotation system. *Genome Res.*, **14**, 942–950.

EGASP (2005) Gene prediction workshop. http://genome.imim.es/gencode/workshop2005.html

Flicek,P. *et al*. (2003) Leveraging the mouse genome for gene prediction in human: from whole-genome shotgun reads to a global synteny map. *Genome Res.*, **13**, 46–54.

Guigo,R. (1998) Assembling genes from predicted exons in linear time with dynamic programming. *J. Comput. Biol.*, **5**, 681–702.

International Human Genome Sequencing Consortium (2001) Initial sequencing and analysis of the human genome. *Nature*, **409**, 860–921.

Kent,W.J. (2002) BLAT—the BLAST-like alignment tool. *Genome Res.*, **12**, 656–664.

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**, 71–74.

Loftus,B.J. *et al*. (2005) The genome of the basidiomycetous yeast and human pathogen Cryptococcus neoformans. *Science*, **307**, 1321–1324.

Majoros,W.H. *et al*. (2004) TigrScan and GlimmerHMM: two open source *ab initio* eukaryotic gene-finders. *Bioinformatics*, **20**, 2878–2879.

Murthy,S.K. *et al*. (1994) A system for induction of oblique decision trees. *J. Artif. Intell. Res.*, **2**, 1–32.

Parra,G. *et al*. (2003) Comparative gene prediction in human and mouse. *Genome Res.*, **13**, 108–117.

Pruitt,K.D. *et al*. (2005) NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.*, **1**, 501–504.

Salzberg,S.L. *et al*. (1999) Interpolated markov models for eukaryotic gene finding. *Genomics*, **59**, 24–31.

Sarawagi,S. and Cohen,W.W. (2004) Semi-markov conditional random fields for information extraction. In *Proceedings of the Advances in Neural Information Processing Systems*, 17 (NIPS 2004), Vancourer, BC, Canada.

Siepel,A. and Haussler,D. (2003) Combining phylogenetic and hidden markov models in biosequence analysis. In *Proceedings of the 7th Annual International Conference on Computational Molecular Biology (RECOMB 2003)*, Berlin, Germany, pp. 277–286.

The ENCODE Project Consortium (2004) The ENCODE (ENCyclopedia of DNA elements) project. *Science*, **306**, 636–640.

Venter,J.C. *et al*. (2001) The sequence of the human genome. *Science*, **291**, 1304–1351.

Wheeler,D.L. *et al*. (2003) Database resources of the national center for biotechnology. *Nucleic Acids Res.*, **31**, 28–33.