International Journal of Biological, Life and Agricultural Sciences
ISSN: 2415-6612
Vol:1, No:1, 2007

Computing Entropy for Ortholog Detection

Hsing-Kuo Pao and John Case

Abstract— **Biological sequences from different species are called** *orthologs* **if they evolved from a sequence of a common ancestor species and they have the same biological function. Approximations of Kolmogorov complexity or entropy of biological sequences are already well known to be useful in extracting** *similarity* **information between such sequences — in the interest, for example, of ortholog detection. As is well known, the exact Kolmogorov complexity is not algorithmically computable. In practice one can approximate it by computable compression methods.** *However*, **such compression methods do not provide a good approximation to Kolmogorov complexity for** *short* **sequences. Herein is suggested a new approach to overcome the problem that compression approximations may not work well on short sequences. This approach is inspired by new,** *conditional* **computations of Kolmogorov entropy. A main contribution of the empirical work described shows the new set of entropy-based machine learning attributes provides good separation between positive (ortholog) and negative (non-ortholog) data — better than with good, previously known alternatives (which do not employ some means to handle short sequences well). Also empirically compared are the new entropy based attribute set and a number of other, more standard similarity attributes sets commonly used in genomic analysis. The various similarity attributes are evaluated by cross validation, through boosted decision tree induction** *C5.0*, **and by Receiver Operating Characteristic (ROC) analysis. The results point to the conclusion: the new, entropy based attribute set by itself is not the one giving the best prediction; however, it is the best attribute set for use in improving the other, standard attribute sets when conjoined with them.**

Keywords—**compression, decision tree, entropy, ortholog, ROC.**

I. Introduction

We consider *Kolmogorov entropy* (or complexity) analysis [1] for biological sequence comparison. One of our goals is to calculate approximate entropy in a new and useful way regarding two or more sequences and their corresponding species to serve as the basis of some new, machine learning attributes, the latter to be used as an aid in detecting orthology. Particularly, we can deal with relatively *short* sequences which are usually difficult, either from the entropy estimation or ortholog detection viewpoint. Short sequences have too few codewords to establish patterns for entropy estimation and usually create unavoided bias [2]. Particularly for distantly related sequences, short ones have few evidences of matches to be really separated from purely random false positives [3]. Our results show good separation between positives and negatives — compared to good standard methods of entropy estimation. Under a machine learning framework, employed for comparative purposes, are a number of attributes sets each based on one of several *standard* sequence alignment methods. In the present paper, then, besides introducing our new, approximate entropy based attributes, we comparatively evaluate all these similarity attributes.[1] We empirically show our new entropy based attributes are competitive with the other similarity attribute sets *and* enhance orthology detection. The evaluation techniques we employ are: 1. decision tree induction from [5] with and without a variant of *AdaBoost* [6] (as implemented in Quinlan's *C5.0*) and cross-validation [7] [8], and 2. ROC analysis for comparing Area Under the Curve (AUC) [9]. For cross-validation of boosted decision tree induction, considered are both the results of: 1. employing essentially only an attribute set being evaluated, and 2. employing all *except* an attribute set being evaluated.

Similarity based on entropy calculation can be understood to be *segment similarity* where the segments can be combinatorially rearranged. Global alignment techniques, such as Gotoh's variant [10] of [11] or *ClustalW* [12][2], and local ones, such as the local alignment from [13] or *Matcher* [14], by contrast, keep matching segments in order, and may have a problem matching ABC and ACB, for long segments B & C. Standard alignment techniques require assigning *constant* gap penalties and, therefore, may have a problem aligning sequences $ABCD$ and ABD, where C is long. Arguably, to assume constant gap penalties among different species or even among different sequences in a given species is not appropriate in general. The segment-to-segment technique *DIALIGN* by [15] [16] is intermediate: without any pre-defined gap penalties, it can handle matching $ABCD$ and ABD (with a long C); but cannot handle completely matching ABC and ACB.

The entropy or the algorithmic entropy of a finite object is defined as the length of the shortest computer program to output this object. Because this entropy value cannot be computed in general [1], we adopt various compression methods to *approximate* the value. We examine two (of many) compressors, the UNIX *gzip* based on the well-known algorithm *LZ77* from [17] and a compressor specially designed for genomic data, *GenCompress*, from [18]. We discuss in sections below several possible corresponding formulae to use in attribute selection.

A. Applied Framework

More specifically, regarding the species featured in our data, our problem is to find correspondent chicken orthologs for a given human-mouse orthologous pair, and particularly difficult (but important for our intended application in agriculture) are the immune function cases which tend to be highly divergent between species *and to be relatively short* [4]. The positive training data are of the form of (X_c, X_h, X_m), where X_c, X_h and X_m are orthologous chicken, human and mouse sequences, respectively. Regarding the negative training data, see Sec. V.

II. Previous Work

Researchers have worked in entropy estimation for biological sequences, either by computing frequency of n-mers for

Research supported by United States Department of Agriculture (USDA) Grant #01-04145 at the University of Delaware.

Hsing-Kuo Pao is with Dept. of CSIE, National Taiwan University of Science & Technology, Taipei, Taiwan (phone: +886-2-27301065, E-mail: pao@mail.ntust.edu.tw).

John Case is with Dept. of CIS, University of Delaware, Newark, DE 19716, USA (phone: +1-302-831-2714, E-mail: case@cis.udel.edu).

[1] Generally, orthologous sequences must exhibit *some* sort of similarity. In [4] also exploited are differences, but, in the present paper, we consider only various kinds of similarity.

[2] The first of these two global alignment methods consider only two sequences at a time, but the third can usefully consider two or more sequences at a time.

long enough inputs, called Shannon entropy [19], or by adopting compression methods to obtain an upper bound on entropy [20]. [21] [22] introduced a compression method *CDNA* by considering inexact match in finding patterns. Importantly, [2] improved the compression further by exploiting the reverse complement property of DNA sequences. Also, this latter method produces a good estimation of entropy, e.g., the estimate approaches the actual entropy *for long enough input*. [23] proposed a distance function (d described below) with nice properties for cluster related sequences. They obtained good results, but, for our need to deal (also and in particular) with relatively short sequences, their formula, when approximated by compression formula is not so helpful. In our evaluation below, we work with a simple variant of their formula[3] (yielding the "distance" function D' below) and with our own modification of this formula (yielding the distance function D below) and can show D works better than D'. Our D was explicitly created to handle the case of short sequences.[4]

We choose a number of standard alignment methods from many more possibilities for our comparative benchmarks: global alignment methods from Gotoh [10] and *ClustalW* [12]; the local alignment method from *Matcher* [14]; and, for segment-to-segment alignment (without rearrangement), we choose *DIALIGN* by [15], [16].[5]

III. Entropy and String Compression

Given a compression method and a string s, we approximate the entropy $K(s)$ of the string s by the length $g(s)$ of the compressed version of s.

We discuss next the entropy of the concatenation of two strings s and t. This combined entropy of s and t is approximated by $g(st)$, the length of the compressed string from st. It is known, from [1], that $K(st)$ and $K(ts)$ will share similar values, up to an additive constant. Hence, $g(st)$ is expected to be similar to $g(ts)$.

We also require conditional entropy $K(t|s)$, defined by the shortest program to compute the string t, given the string s for free. The relation between the concatenation entropy and conditional entropy [1] inspires the approximation (correct up to a lower order term),

$$K(t|s) \sim K(st) - K(s). \tag{1}$$

Hence, we use $g(t|s) \sim g(st) - g(s)$ to approximate the conditional entropy $K(t|s)$.

For our entropy attribute set we compress the nucleotide (or NT) sequences and the amino acid (or AA) sequences, for each of three different species, chicken, human, and mouse.

[3] *This* variant is as good for machine learning attributes as their original, and is helpful for understanding *our* new variant.

[4] We confine our study to comparing D' and D since there are so many approaches, including additionally, [24] [25] [26] — each also appropriate only for long sequences.

[5] Our purpose is *not* to explore as many attributes as possible from various alignment methods, but to seek attributes based on "different" types of methods. We would expect, from the use of multiple *disparate* methods, a better chance of separation between positives and negatives, but that similar types of alignment methods would give *relatively* similar separation results to one another. Similarly, we do not investigate all prior entropy estimators other than *gzip* and *GenCompress* to calculate new attributes.

Roughly, the algorithms of *gzip* and *GenCompress* look for repeated strings (for feasibility, repeated strings of some restricted length) and replace each repeat with a reference to the first occurrence. However, *GenCompress* importantly employs approximate instead of exact matching for determining repeats and also looks for (approximate) reverse complements for NT sequences. This looking for matching (exact or approximate) and repeats is why we say above that our entropy based attributes are based on segment similarity where the segments can be combinatorially rearranged.

IV. Entropy Attributes for Ortholog Detection

For two sequences, their concatenation is likely to be highly compressible, *if* a certain percentage of segment similarities exists between them. Our *new* formula for the "distance" between sequences s and t is as follows.

$$D(s,t) = \frac{K(st)}{K(s|\mathbf{S}) + K(t|\mathbf{T})}, \tag{2}$$

where in the numerator, we compute the entropy for the concatenation of the two sequences s and t from different species (either both NT or both AA), and in the denominator, we compute two conditional entropies. The long sequence $\mathbf{S}$ is the result of concatenating together all the sequences in our data set *except* s for the species that sequence s belongs to. $\mathbf{T}$ is similarly based on the sequences except t from t's species.

There are 565 triples of orthologs in our data set.[6] For each ortholog, we produce three of the long sequences, one for each of the species, chicken, human, and mouse (minus that species' gene for the ortholog). Hence, each long sequence for a given ortholog is the concatenation of $565 - 1 = 564$ sequences.[7] For example, if s denotes a gene sequence from chicken and $\mathbf{S}$ is the long chicken sequence combined from all but the chicken sequence s, the conditional entropy $K(s|\mathbf{S})$ measures the entropy in s given for free knowledge of the rest of the chicken genome (in our data set). There are two reasons for us to use this conditional computation: 1. Compression usually gives bad performance for short strings — any encoding (with the output of constant length blocks) may produce blocks with length close to or even larger than the length of the original data — the long sequences will have a better compression rate than the short sequences, and our data set consists of sequences with varied lengths, and, therefore, our conditional computation can assuage bias between sequences with different lengths; *and* 2. There are always certain common regions between different species, e.g., regions with high $G + C$ content in NT sequences — such regions' similarity has nothing to do with orthology, so we want to remove it from consideration.[8]

Without the conditional calculations, D in Eq. 2 above becomes D' just below.

$$D'(s,t) = \frac{K(st)}{K(s) + K(t)}. \tag{3}$$

[6] Available at http://www.ccl.rutgers.edu/~ouyang/CHM/byName.html, curated as described in [27].

[7] As noted above, we essentially need not worry about the order of single sequences in concatenation for the long sequence for each species.

[8] Such conditionals in the numerator would not make sense and do not provide useful or reasonable attributes.

The formula for D' is, in turn , a useful, approximate *algebraic variant* of the distance function given by [23], i.e.,

$$d(s,t) = 1 - \frac{K(s) - K(s|t)}{K(st)} \sim \frac{2K(st) - K(s) - K(t)}{K(st)} .$$

By Eq. 1, Eq. 3 is essentially the reciprocal of "$2 - d()$". Therefore, we can use Eq. 3 as a representative to do further comparisons between Eq. 2 and $d()$ from [23], if each is used as an attribute in a classification task.

V. Experimental Results

An outcome of our experiments is that the attributes from entropy calculation are important for ortholog predictions. As noted above, the training is done with triples from all three species, chicken and the two mammals human and mouse. We employ *two* mammals (human and mouse) in place of one for extra help from the resultant "triangularization" of data. It it inspired by the idea that multiple sequence alignment usually performs better than pairwise alignment for more evidences of matches.

After computations from entropy/compression and the several different alignment methods (plus *class* information discussed just below), 47 attributes are compiled for each triple. We have four attributes from Gotoh's global alignment, 20 attributes from *DIALIGN*, six attributes from *ClustalW*, 12 attributes from *Matcher*, and four attributes from each compression method.[9] Also, we have one attribute *class* describing the biological function *class*, with six different (discrete) values, e.g., defense or immune system [28][10]; see also [29]. The other five categories are CD(cell division), CS(cell signaling), SM(cell structure), GPE(gene, protein expression) and M(metabolism) of the two mammals. The biological function can be unknown for the chicken counterpart, i.e., for the target sequence. Other 46 attributes are continuous attributes.

Regarding the compression approximation of the algorithmic entropy, as noted above, we employ two methods, *gzip* (result not shown) and *GenCompress*. For comparison, the size of the long, all-but-one-combined sequences is around $800,000$ bases, and the size of a single gene varies from a few hundred to a few thousand bases.

For *C5.0* prediction, the vectorized data with the (up to) 47 attributes are used for training. Apart from the collection of 565 orthologs, namely, the positive data, we need to obtain a set of *negative* data for a complete training set for classification. For ROC analysis too, negative data are necessary for the construction of ROC curves. We employ a criterion similar to that of [4] for the negative data generation. We collect the negatives from each possible combination of (X_c, Y_h, Y_m), where Y_h, Y_m are othologous, but X_c is not orthologous to Y_h, Y_m. *Additionally*, from such a big set, we remove certain "easy" triples either whose classifications are trivial (e.g., non-orthologous due to big difference between sequence lengths) or which are considered not informative, in the sense of building a classifier (e.g., too low sequence similarity). There are, then, 6075 negative data in total, meeting these requirements [4].

To evaluate how our entropy based attribute set is superior to other attributes, we adopt three methods:

1. Cross-validation: The ultimate goal for machine learning applications is to predict unknowns. We use cross-validation with *C5.0* to judge the extent that one attribute set is more useful than another in this regard. This is discussed further in Sec. V-A. N.B. The Standard Error in the various percent errors reported is $< 0.05\%$.
2. Salience in decision tree: For *C5.0*'s *first/best* decision tree, the attribute whose value is tested on the top explains more data than those attributes further down the tree. This provides a criterion to separate salient attributes from not-so-salient ones. Our result shows *(one of) the entropy based attribute(s) is always on top, while included in the attribute set* (trees not shown).
3. ROC analysis: The ROC curve is commonly used in diagnostic research. A measure computing the area under the ROC curve (AUC) is frequently used as a criterion to see if one attribute is more useful than another. This will be further discussed in Sec. V-B.

For the various evaluation methods, we carried out several series of experiments each based on comparing various attribute sets.

A. Cross-validation

For space limitation we omit details re our first two series of cross-validation experiments showing, re ortholog detection, superiority of D to D' and *GenCompress* to *gzip*. In each experiment of the next two series (A, B below) we employ *C5.0* with 10-tree boosting and carry out 10-fold cross-validation, with 30 repeats. The *class* information is included in *each* experiment, where it serves as a background attribute.

A. The point of this series (of five experiments) is to compare various similarity assessing attribute sets with one another *by employing each such attribute set one at a time* (*class* is always included). For each experiment we choose one attribute set from a single one of the five similarity assessing methods as *the* set of attributes to use. Each chosen attribute set will have size depending on which similarity assessing method it corresponds to. For instance, there are six (plus one) attributes in the set if we choose *ClustalW* but $12 + 1$ attributes in the set if we choose *Matcher*.

B. The point of this series (of six experiments) is to compare five similarity assessing attribute sets with one another *by evaluating the effect of leaving up to one out* (with the *class* attribute always in). In this series, the same five similarity assessing attribute sets are employed as in A. However, we leave up to one such set out. This is to see how, if at all, *C5.0*'s performance is weakened by each removal.

In Tab. I, for the series, A, the entropy/compression attribute set gives the third best prediction among all similarity assessing attribute sets. The attribute set derived from *ClustalW* is the one, when by itself, yields the best prediction, and this with only six (plus one) attributes in its attribute set.

In the series, B, it can be seen that the entropy/compression based attribute set dominates the overall prediction. Wit*h*o*ut* that single attribute set, we have the worst prediction rate.

[9] E.g., the four attributes per compression method are from NT vs. AA sequences combined with comparing chicken to mouse vs. human separately. Due to space limitations, we can not describe in detail the attribute set that goes with each other similarity assessing method.

[10] http://tigr.org/docs/tigr-scripts/egad_scripts/role_report.spl

TABLE I
RESULTS OF 10-FOLD CROSS-VALIDATION, COMPARING THE FIVE SIMILARITY ASSESSING ATTRIBUTE SETS.

Attribute Set	**Missed in Testing: +/-(%)**	
	A: only one in (& *class*)	B: at most one out
w. all attr.	–	41.9 / 15.6 (0.87%)
Matcher	(w.) 71.7 / 53.7 (1.89%)	(w/o) 41.3 / 16.3 (0.87%)
Gotoh	(w.) 70.8 / 67.7 (2.09%)	(w/o) 43.2 / 15.0 (0.88%)
ClustalW	(w.) 45.3 / 46.1 (1.38%)	(w/o) 41.9 / 15.5 (0.87%)
DIALIGN	(w.) 51.7 / 51.8 (1.56%)	(w/o) 40.1 / 16.1 (0.85%)
GenCompress	(w.) **101.6 / 12.6 (1.72%)**	(w/o) **47.9 / 50.3 (1.48%)**

The results of Tab. I together can be understood to mean: *the entropy based attribute set by itself is not the one giving the best prediction; however, it is the best attribute set for use in improving the others when conjoined with them.*

B. ROC Analysis

In the first part of this series, we constructed the ROC curves for *single* individual attributes (e.g., a_i) and computed their AUC [9] to see if one attribute gives more prediction power than another. Given a classifier $a_i > c$ for some constant c predicting positive data (assuming most positives give larger values than most negatives, e.g., as with identities for two aligned sequences), we can construct the associated ROC curve on the unit square $[0,1] \times [0,1]$ by plotting all of the points $(\#(\textit{false pos.})/\#(\text{neg.}), \#(\textit{true pos.})/\#(\text{pos.}))$. The meaning of AUC is the probability of correctly labeling a pair of positive and negative data, through the measuring of a_i.

By measuring the AUC value for all single attributes, the entropy based attributes *do not* give good results compared to attributes from many other standard alignment methods. E.g., for the (AUC) *best* entropy based attribute, $D(C,H)$, our entropy based attribute between AA sequences for chicken and human, we have $\mathrm{AUC} = 0.927$. Whereas, we have $\mathrm{AUC} = 0.994$ for the (AUC) best attribute $\mathrm{id}_{\mathrm{Gotoh}}(c,h)$, Gotoh's percent identity for NT sequences between chicken and human. However, when two attributes are considered simultaneously, the AUC criterion prefers the combination of *an entropy attribute with an attribute from identities*. The AUC value for two attributes is computed by a linear transformation from two dimensions to one, followed by the regular AUC computation (where the projected angle is chosen to maximize the AUC value). Moreover, while the identity attribute can be changed to another standard similarity based attribute without lowering too much the AUC value, the entropy *cannot* be substituted by a non-entropy attribute without significantly lowering the AUC value. Hence, the entropy attribute set is an excellent *helper* for ortholog prediction. We have $\mathrm{AUC} = 0.996$ from the two attributes, $D(C,M)$ and $\mathrm{id}_{\mathrm{Gotoh}}(c,h)$, as the *best* combination. Supported, then, is the same conclusion as from the series A and B above: that *our (best) entropy based attribute set shows its superiority (only) when conjoined with standard similarity assessing attributes*.

REFERENCES

[1] M. Li and P. Vitányi, *An Introduction to Kolmogorov Complexity and Its Applications (2nd Ed.),* Springer, New York, 1997.
[2] J. K. Lanctot, M. Li, and E.-H. Yang, "Estimating DNA sequence entropy," in *Symposium on Discrete Algorithms*, 2000, pp. 409–418.
[3] S. Karlin and S. F. Altschul, "Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes," *Proc. Natl. Acad. Sci. USA*, vol. 87, pp. 2264–2268, 1990.
[4] M. Ouyang, J. Case, and J. Burnside, "Divide and conquer machine learning for a genomics analogy problem (progress report)," in *Discovery Science*, 2001, pp. 290–303.
[5] J. R. Quinlan, *C4.5: Programs for machine learning*, Morgan Kaufmann Publishers, San Mateo, CA, 1993.
[6] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Science*, vol. 55, pp. 119–139, 1997.
[7] F. Mosteller and J. Tukey, *Data Analysis and Regression: A Second Course in Statistics*, Addison-Wesley, Reading, Mass., 1977.
[8] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *Journal of the Royal Statistical Society*, vol. 36, pp. 111–147, 1974.
[9] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol. 143, pp. 29–36, 1982.
[10] O. Gotoh, "An improved algorithm for matching biological sequences," *J. Mol. Biol.*, vol. 162, pp. 705–708, 1982.
[11] S. Needleman and C. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology*, vol. 48, pp. 443–453, 1970.
[12] J. D. Thompson, D. G. Higgins, and T. J. Gibson, "Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting position-specific gap penalties and weight matrix choice," *Nucleic Acids Res.*, vol. 22, no. 22, pp. 4673–4680, 1994.
[13] T. F. Smith and M. S. Waterman, "Comparison of biosequences," *Adv. Appl. Math.*, vol. 2, pp. 482–489, 1981.
[14] X. Huang and W. Miller, "A time efficient, linear space local similarity algorithm," *Adv. Appl. Math.*, vol. 12, pp. 337–357, 1991.
[15] B. Morgenstern, A. Dress, and T. Werner, "Multiple DNA and protein sequence alignment based on segment-to-segment comparison," *Proc. Natl. Acad. Sci. USA*, vol. 93, no. 22, pp. 12098–12103, 1996.
[16] B. Morgenstern, K. Frech, A. Dress, and T. Werner, "Dialign: finding local similarities by multiple sequence alignment," *Bioinformatics*, vol. 14, no. 3, pp. 290–294, 1998.
[17] A. Lempel and J. Ziv, "A universal algorithm for sequential data compression," *IEEE Trans. Inf. Theory*, vol. 23, no. 3, pp. 337–343, 1977.
[18] X. Chen, S. Kwong, and M. Li, "A compression algorithm for DNA sequences and its applications in genome comparison," in *RECOMB*, 2000, p. 107.
[19] C. E. Shannon, "A mathematical theory of communication," *Bell Syst Tech. J.*, vol. 27, pp. 379–423, 1948.
[20] S. Grumbach and F. Tahi, "Compression of DNA sequences," in *Proceedings of the IEEE Symposium on Data Compression*, 1993, pp. 340–350.
[21] D. M. Loewenstern, H. Hirsh, P. Yianilos, and M. Noordewier, "DNA sequence classification using compression-based induction.," Tech. Rep. 95-04, DIMACS, 1995.
[22] D. M. Loewenstern and P. N. Yianilos, "Significantly lower entropy estimates for natural DNA sequences," in *IEEE Data Compression Conf., DCC97*, 1997, pp. 151–160.
[23] M. Li, J. H. Badger, X. Chen, S. Kwong, P. Kearney, and H. Zhang, "An information-based sequence distance and its application to whole mitochondrial genome phylogeny," *Bioinformatics*, vol. 17, no. 2, pp. 149–154, 2001.
[24] D. Benedetto, E. Caglioti, and V. Loreto, "Language trees and zipping," *Physical Review Letters*, vol. 88, no. 4, pp. 048702, 2002.
[25] Ming Li, Xin Chen, Xin Li, Bin Ma, and Paul Vitanyi, "The similarity metric," in *Proc. 14th ACM-SIAM Symp. Discrete Algorithms (SODA)*, 2003.
[26] D. R. Powell, D. L. Dowe, L. Allison, and T. I. Dix, "Discovering simple DNA sequences by compression," in *PSB*. 1998, pp. 597–608, World Scientific.
[27] M. Ouyang, J. Case, V. Tirunagaru, and J. Burnside, "565 triples of chicken, human, and mouse candidate orthologs," *Journal of Molecular Evolution*, vol. 57, pp. 271–281, 2003.
[28] M.D. Adams, A.R. Kerlavage, R.D. Fleischmann, R.A. Fuldner, C.J. Bult, N.H. Lee, E.F. Kirkness, K.G. Weinstock, J.D. Gocayne, O. White, and et al, "Initial assessment of human gene diversity and expression patterns based upon 83 million nucleotides of cDNA sequence," *Nature*, vol. 377, pp. 3–174, 1995.
[29] V. Tirunagaru, L. Sofer, and J. Burnside, "An expressed sequence tag database of activated chicken T cells: Sequence analysis of 5000 cDNA clones," *Genomics*, vol. 66, pp. 144–151, 2000.