The GeneMine system for genome/proteome annotation and collaborative data mining

by C. Lee
K. Irizarry

As genome data and bioinformatics resources grow exponentially in size and complexity, there is an increasing need for software that can bridge the gap between biologists with questions and the worldwide set of highly specialized tools for answering them. The GeneMine system for small- to medium-scale genome analysis provides: (1) automated analysis of DNA (deoxyribonucleic acid) and protein sequence data using over 50 different analysis servers via the Internet, integrating data from homologous functions, tissue expression patterns, mapping, polymorphisms, model organism data and phenotypes, protein structural domains, active sites, motifs and other features, etc., (2) automated filtering and data reduction to highlight significant and interesting patterns, (3) a visual data-mining interface for rapidly exploring correlations, patterns, and contradictions within these data via aggregation, overlay, and drill-down, all projected onto relevant sequence alignments and three-dimensional structures, (4) a plug-in architecture that makes adding new types of analysis, data sources, and servers (including anything on the Internet) as easy as supplying the relevant URLs (uniform resource locators), (5) a hypertext system that lets users create and share "live" views of their discoveries by embedding three-dimensional structures, alignments, and annotation data within their documents, and (6) an integrated database schema for mining large GeneMine data sets in a relational database. The value of the GeneMine system is that it automatically brings together and uncovers important functional information from a much wider range of sources than a given specialist would normally think to query, resulting in insights that the researcher was not planning to look for. In this paper we present the architecture of the software for integrating and mining very diverse biological data, and cross-validation of gene function predictions. The software is freely available at http://www.bioinformatics.ucla.edu/genemine.

The Human Genome Project and related advances in technology have drastically increased the amount of data that can be brought to bear on any biological or medical question. Genomics technologies are providing an almost unmanageably detailed picture of cellular mechanisms and gene functions. They have accelerated traditional molecular biology techniques (e.g., sequencing, Northern blots) by real factors of 100- to 100 000-fold. This data explosion poses major challenges for data mining, both in terms of the sheer mass and complexity of the data and the sophisticated scientific questions that must be asked to make important discoveries. In this paper we analyze the specific data-mining problems characteristic of bioinformatics by means of our experiences in developing and refining GeneMine, a software system for analysis of gene functions.[1–3]

First, it is essential to understand the technical and cultural background of biological research. Biology is an extremely diverse discipline, broken into many specialties. Until recently, biology experiments required much human labor for each unit of data. Researchers had to analyze individual results by hand, because they had little familiarity with data-mining methodologies. Most data were either not archived in any database or stored in one of many incompatible databases. In this fragmented and heterogeneous

0018-8670/01/$5.00 © 2001 IBM

environment, the Web emerged as the dominant model for bioinformatics, in which Web pages present the set of possible queries from which the (knowledgeable) user can find the proper query to answer a given question. In this "expert query" model, a bioinformatics expert may be able to think of the proper query, decide if it is worth the effort, and decipher the complex and often voluminous results. A less-expert user might not even be aware of what queries are possible. Although this Web model made data much more broadly available and useful, the basic modes of querying the data (e.g., BLAST, the Basic Local Alignment Search Tool[4]) have remained largely unchanged. Whereas genomics data have made dramatic orders-of-magnitude advances, most biologists are querying and analyzing those data in much the same ways they did five years ago.

This paper analyzes these challenges and our resulting design choices for GeneMine in three broad areas. First, we opted to focus on information visualization, providing an interactive, visual tool for human scientists to make and validate discoveries, as opposed to automated data-mining programs for *computers* to make discoveries (e.g., Bayesian methods for polymorphism discovery[5,6]). GeneMine is designed to assist scientific inference from multiple lines of evidence for problems that still require human intelligence. At this early stage in bioinformatics, most real questions have this character. Such problems demand an exploratory tool that exposes patterns to the scientist's perception and facilitates rapid exploration of hypotheses. Second, GeneMine deploys a client-side approach to heterogeneous data integration, as opposed to heavier-weight server-side strategies used in many other successful systems.[7–9] The client-side approach fits especially well to a visual, interactive tool; we discuss its advantages and disadvantages. Third, given biologists' unfamiliarity with data mining and bioinformatics, we decided to use an information *push* model instead of the conventional *pull* mindset assumed in Web or database query systems. GeneMine uses query and data-mining automation to *push* relevant information from many sources into the users' view for the specific genes on which they are working. We discuss the layers of automation and data-mining techniques used to achieve this (data aggregation and filtering, drill-down, cross-validation). This trio of design choices made GeneMine unique in bioinformatics at the time of its development (1993), and its lessons may be relevant to other problem domains with similar characteristics.

The problem: Discovering and validating gene function

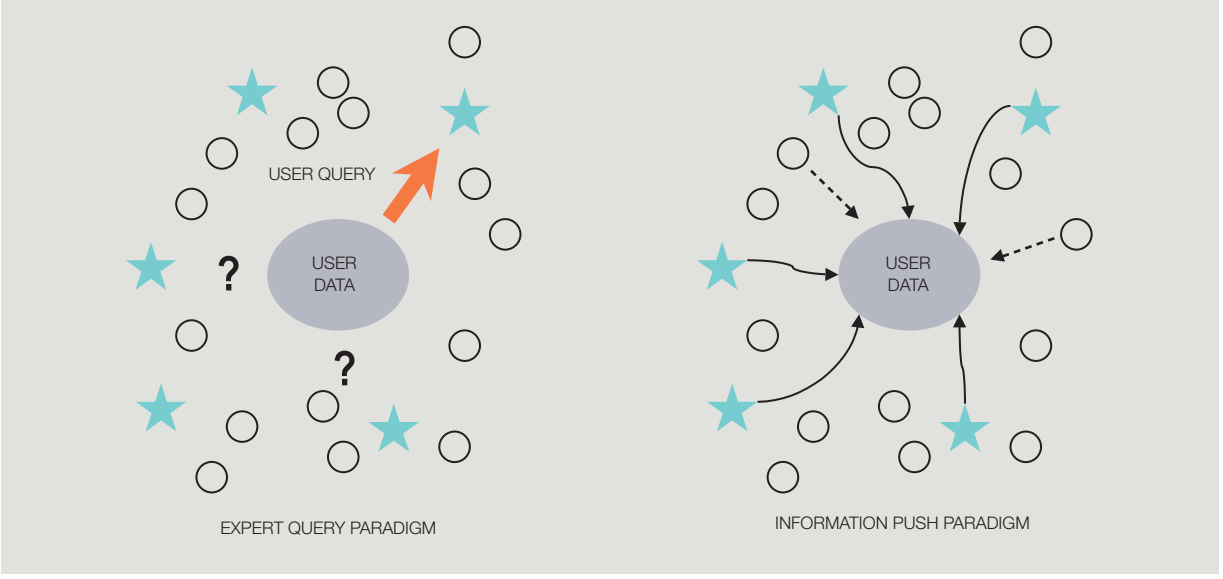
The explosion in genome sequencing (more than 60 completed as of November 2000) has created a massive supply of new genes whose function must be inferred to discover which are involved in human disease and to address other questions of enormous medical and economic importance. Until recently, only about 10–20 percent of human genes had been identified (over the previous 50 years) and studied sufficiently even to be given a scientific name. In the last two years researchers have obtained the remaining 90 percent of genes, but do not know their functions.

GeneMine was designed to help scientists rapidly infer, validate, and propose experimental tests for the likely functions of unknown genes. It deploys classic data-mining techniques such as association rules, data generalization, and classification or clustering. However, the role of the human scientist in perceiving subtle patterns and formulating a complex scientific hypothesis is paramount. Thus, GeneMine was designed not to replace the scientist in data mining, but rather to empower the scientist's ability to perceive, explore, and test ideas rapidly.

The need for an information push paradigm

In the "expert query" model that typifies bioinformatics on the Web and database query languages, results can only be obtained if the user actively thinks of the appropriate query. Many results, especially surprising results from unexpected sources, are missed because the user never thinks of asking the question, or dismisses it as unlikely to be worth the effort. Most importantly, biologists are less likely to cross the boundaries of their highly specialized fields to look for useful queries, either because they are unaware of what is possible or because they worry about interpreting the significance of the results. In an era of complete genome sequences, one of the greatest opportunities lies in integrating information from multiple organisms that can provide very complementary kinds of functional data, e.g., yeast (gene knock-outs, two-hybrid data, etc.), *Drosophila* and *C. elegans* (developmental mutants, phenotypes), mouse (genetics, animal models of disease), and human (human diseases, genomic mapping, etc.). The barriers of expertise and the fragmentation of bioinformatics tools and databases discourage scientists from exploiting this important opportunity.

Figure 1 Information push model of GeneMine that automates a conceptual set of queries to push relevant information into the user's view (solid lines), while filtering out irrelevant or redundant data (dashed lines)



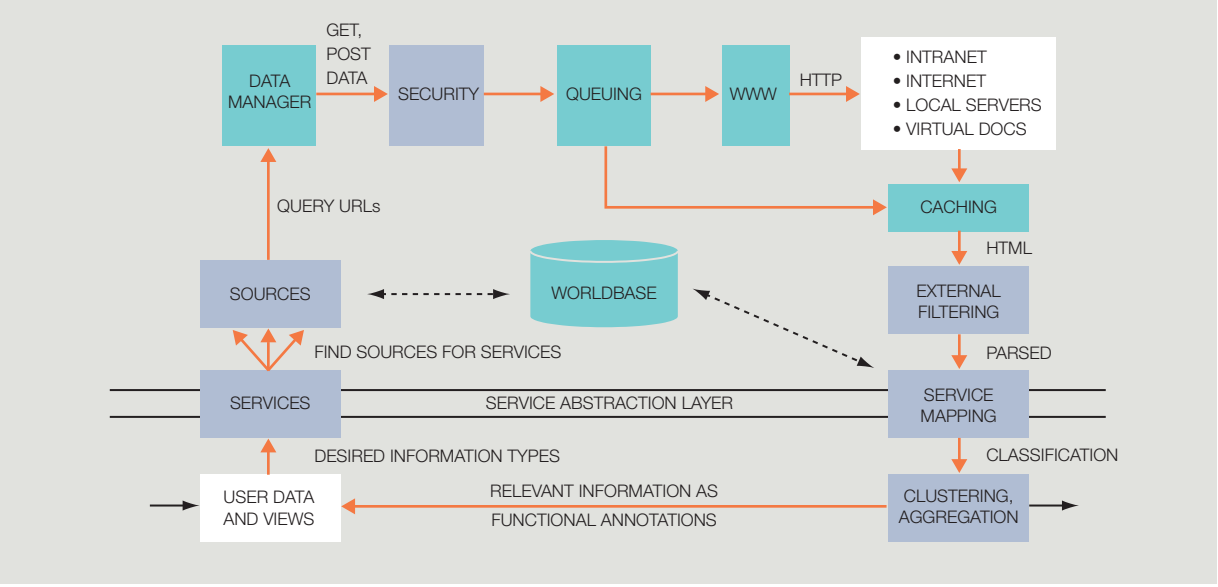
GeneMine transforms this "expert query" problem into a visualization and data-mining problem (Figure 1). It acts as an agent collecting and filtering relevant data from diverse databases, analysis programs, and servers. By using automation to run all possible queries on each new gene sequence introduced by the user, GeneMine filters the results for significant and interesting patterns and presents them in a highly distilled form that can be explored deeply. Although this may initially seem wasteful of CPU power, usage data show that the limiting resource is usually not CPU time, but rather *human time*. Collecting these diverse data manually on the Web takes so much time that scientists' sensible instinct for efficiency makes them reluctant to perform queries that do not have a specific expected result. However, the real power of diverse bioinformatics data and analyses is not to confirm existing expectations, but rather to provide new, unexpected connections and insights. The benefit of this "information push" model over the "expert query" model is not only fast and effortless answers to questions the scientist already has in mind, but also insights into patterns that the scientist had not even thought to look for. The effect is one of moving from an information-poor environment to an information-rich environment.

A client-side architecture for heterogeneous data integration

Bioinformatics databases and services are highly complex, heterogeneous, fragmented, and frequently incompatible. Yet the major need is for systems that can integrate these diverse data to make discoveries. How do we achieve this? Broadly speaking, heterogeneous database integration can be implemented via server-side or client-side architectures. In bioinformatics a number of server-side architectures have been described, including databases that seek to integrate diverse data[7,10] and federated database models based on metadescriptions of their component database schemas.[8,9]

Because GeneMine has a different focus on providing an interactive, push client for visual data mining, we opted for a client-side integration strategy. This strategy had advantages of high interactivity (low latency, because most roll-up and drill-down operations can be performed directly on the client, rather than invoking queries to one or more servers) and strong integration of heterogeneous data in the user interface. It also allowed us to use a much "lighter weight" architecture for integration that is much simpler in both its implementation and usage,

Figure 2 Architecture and information processing flow of the automatic annotation of GeneMine



and which increases the scope of data sources that can be easily integrated to encompass the entire Web. Rather than requiring specialized architectures and protocols such as Kleisli,[9] OPM (Object Protocol Model), or CORBA** (Component Object Request Broker Architecture**)[11] and detailed metaschema information for the whole federation, GeneMine uses lightweight protocols such as HTTP (HyperText Transfer Protocol) to connect with any server on the Web and does not require schemas. The main disadvantage is its lightweight relationship to servers: Because it does not require taking control of server schemas, it is vulnerable to server-side changes in schema and presentation—a common event on the Web.

GeneMine catalogs servers in a simple "database of databases" (Worldbase, seen in Figure 2) that completely encapsulates its lightweight integration module. This novel approach has several benefits. Using this plug-in architecture, any server form on the Web (anywhere a Web browser can enter data for analysis) can be added by providing its URL (uniform resource locator) and other simple information. A new server can be added in minutes, without modifying the server to adhere to a new protocol, or without even knowledge of its schema. There are few limitations on services (e.g., Web-based query forms, relational databases, and local executables), as long as a client-server connection is possible (Figure 2). In this respect GeneMine is similar to several other systems that have adopted a "lightweight" approach to heterogeneous database integration, such as MAGPIE,[12] PEDANT (Protein Extraction, Description, and ANalysis Tool),[13] and GeneQuiz.[14] With minimal effort we have integrated dozens of services (see Table 1, later, for examples), compared with the handful typically integrated by heavyweight server-side projects.[9,15] The parallel architecture that GeneMine has for performing its many queries generates results in real time, typically initial results within 2–3 seconds. Because GeneMine acts as an information client coordinating many simultaneous analyses over a large number of servers, it naturally breaks up the gene annotation process with a high degree of parallel processing. The main disadvantage is dependence on external databases and servers.

Another innovative feature of this architecture was its organization of heterogeneous databases into abstract data types called *services*. This feature allowed us to "program the Web" by creating an abstract application programming interface (API) to its capabilities, in the form of *service* paths instead of hard-coded URLs. The pending proposal by the Internet Engineering Task Force (IETF) of uniform resource names (URNs) is similar in concept (not yet adopted in common Web browsers or servers). Each individ-

ual plug-in can be associated with one or more named services, a classification scheme that abstracts what kind of information the plug-in produces. GeneMine extends the definition of file paths to include not only URLs but also abstract service names (with data as arguments); thus, queries can be composed directly in terms of what data are desired, rather than in terms of the location or path (e.g., URL). One simple benefit of this definition is automatic rollover: If a given server fails, GeneMine automatically rolls over to a backup server registered to the same named service. Rollover is a natural outgrowth of GeneMine's position as a client: it cannot assume that the servers will actually work.

Service mapping. GeneMine analyzes paths embedded in returned data and recognizes those that map back to known data services. It automatically remaps these paths into the abstract service form, since this makes available all the benefits of GeneMine's knowledge of what can be done specifically with that kind of data.

Filtering. Since different servers return data in heterogeneous formats, GeneMine provides an open mechanism for reformatting the data into a simple line format, using *Perl*, *awk*, or any other tool. This mechanism makes it quite straightforward to plug in any new source without limitations of format.

Chaining and recursion. A series of independent filter modules can be chained arbitrarily or can return new queries (as URLs or service requests) for GeneMine to perform recursively.

Caching. GeneMine caches many of its results to avoid performing the exact same query redundantly within a short period.

Batch processing. The automatic annotation pipeline can operate either as part of an interactive data visualization application (i.e., GeneMine) or can be run on a large set of data from the command line, outputting results on standard out or transmitted to a relational database. Once stored in a relational database, GeneMine's annotations over an entire database of sequences can be queried for particular patterns of overlap between desired categories of functional information, and the results viewed with the GeneMine visualization client.

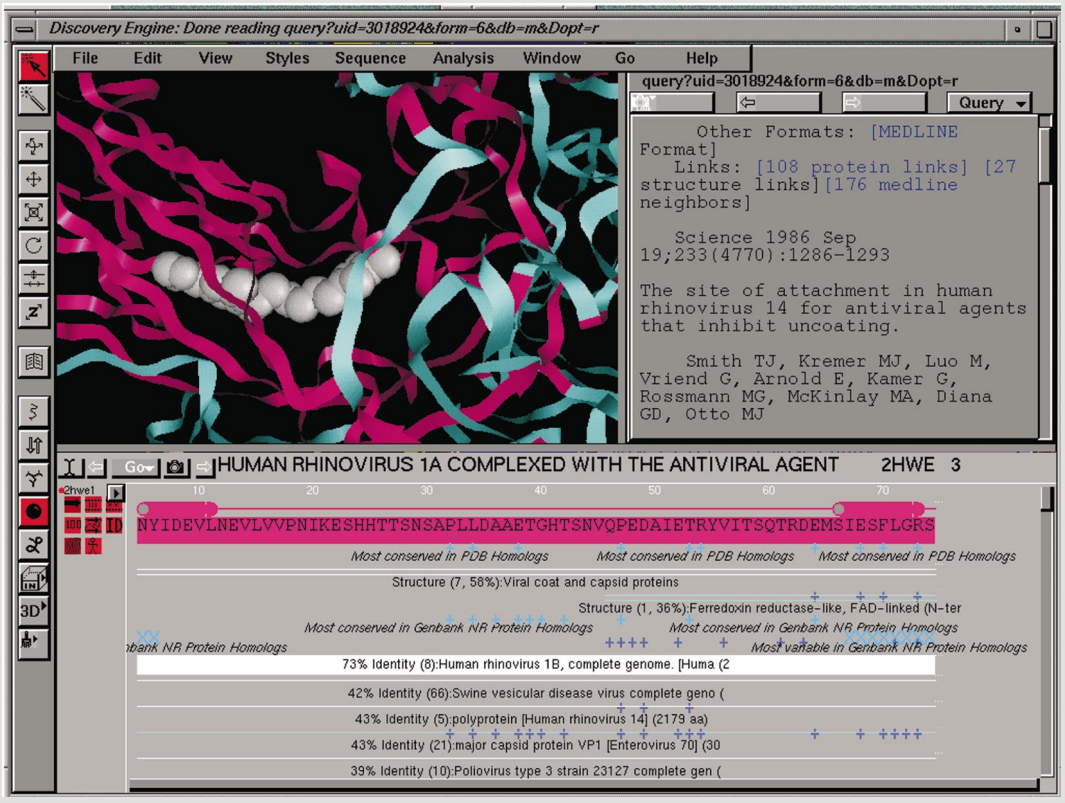
An interactive visual data-mining tool

Bioinformatics poses special challenges to data-mining methods.[15] Here we analyze the application of classic approaches such as association rules, decision trees, clustering, and multidimensional analysis[16] to bioinformatics problems.[17,18] A key issue in biological databases is their extreme diversity and the enormous breadth of data types, ranging from clinical patient databases and population genetics data to genome sequence and expression data, to chemical structure and activity databases (to name just a few). Seeking association rules in such a complex data set scales as $O(N^2)$ or worse when one considers all possible pairs of data types or higher-order combinations. Fortunately, this galaxy of data types can be resolved into a simple star topology by taking advantage of the central role of DNA (dioxyribonucleic acid) in biology. Since nearly all biological activities, structures, and properties derive from one or more genes, these data can be reorganized to use the gene sequence as a "hub" that connects all the diverse data.

The visual data-mining environment of GeneMine is organized on this principle (Figure 3). Its *sequence window* seen at the bottom of the figure is the central starting point for nearly all analysis, where DNA and protein sequences, as well as annotations from GeneMine, are shown. Around this window are the *structure window* at upper left (for three-dimensional atomic structure and molecular modeling[19–21]) *functional annotations* (function features associated with specific residue(s) of sequences), and the *information window* at upper right (for drill-down, browsing, and user hypertext documents containing embedded views of the three previous kinds of data). These views are completely interconnected; any action in one is reflected in all, permitting users to perceive and explore the detailed association rules of all the data through their interconnection in the gene sequence. This fully integrated "information hub" design was a key innovation in GeneMine (starting from the earliest version called "LOOK" in 1993) and has been recently adopted by other systems such as Cn3D.[22] In addition, the sequential nature of DNA and protein makes time-series data-mining techniques very important.

As shown in Figure 3, any macromolecule structure appearing in the structure window is also represented as a DNA or protein sequence in the sequence window. Any number of structures can be shown, superimposed, and analyzed simultaneously. Any number of sequences can be shown at once; they are automatically aligned by dynamic programming.[23,24] Annotations (e.g., in Figure 3, the secondary structure of the protein; its homologies to other protein

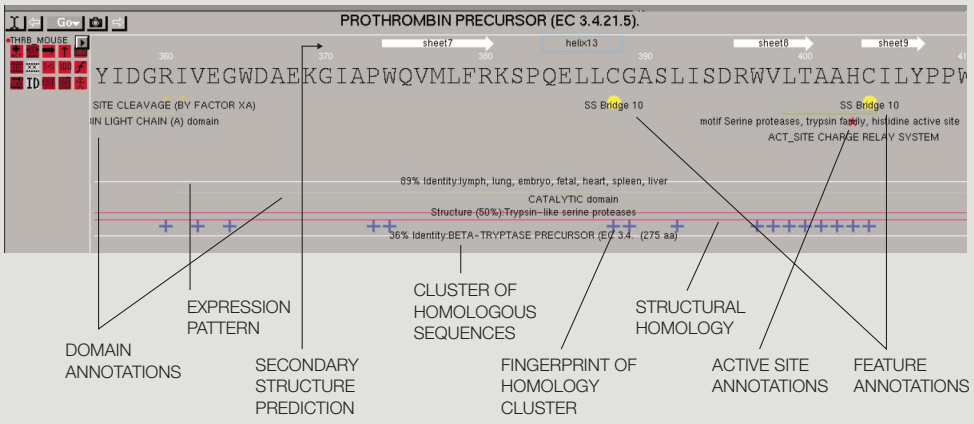
Figure 3 A view of the GeneMine application (displaying the rhinovirus 1a crystal structure 2HWE [PDB])



familes, etc.) are shown above and below the sequence residues with which they are associated. An annotation can mark an entire sequence, discontiguous regions, or even individual residue(s).

GeneMine is also unique in the range of information types that it draws together in an integrated visual mining interface, using the *annotation* metaphor to attach any kind of information to specific locations in a sequence. Because nearly all biological activities ultimately attach to an individual gene, and frequently to a specific range or individual residue within that gene, annotations provide a very general mechanism for integrating extremely diverse types of data from different sources (Figure 4 and Table 1). In Figure 4, the distinct types of data found for a given sequence are displayed as small icons beneath the name of the sequence (at left). The user can turn the display of each annotation type on or off by clicking the icon or show or hide individual annotations in any combination. A critical design goal was to include information all the way from the genomic DNA level (the blueprint for the organism) to its expression as working proteins, with their complex structure-function relationships. The current annotation types of GeneMine include genetic features (e.g., physical or genetic mapping, polymorphisms, open-reading frames, exons or introns), protein structure features (e.g., domains, secondary structure, disulfide bridges), functional features (active site or binding site residues, functional motifs), homology relationships or patterns, gene expression information,

Figure 4 GeneMine integrates many types of information from diverse sources as "annotations" attached to specific residues or regions of each sequence



model organism data, disease associations, and literature links from SWISS-PROT (protein database developed in Switzerland), GenBank, PROSITE (database of protein families and domains), OMIM (Online Mendelian Inheritance in Man), FlyBase, and other databases. The ability to integrate these diverse data draws not only on the advantages of the lightweight client architecture of GeneMine but also on the tremendous efforts throughout the bioinformatics community to make data available via the Web (e.g., Table 1[25–35]).

A number of bioinformatics data-mining systems, such as MAGPIE,[12] PEDANT,[13] and GeneQuiz,[14] have integrated different aspects of these types of data but are server-side architectures that solve different problems than GeneMine. GeneQuiz makes completely automatic gene function predictions and stores them in a database. User interaction is limited to HTML (HyperText Markup Language) views of the output tables (using a Web browser as the interface). By contrast, GeneMine does not try to make an automatic function prediction but seeks instead to provide a visual data-mining tool for a human scientist to make his or her own inferences. MAGPIE is another server-side solution that provides users with HTML tables that can be viewed in a Web browser. It concentrates on genome sequencing project management, ORF (Open Reading Frame) identification, and metabolic pathways, in contrast to GeneMine's focus on gene or protein structure-function. PEDANT also employs a server-side architecture: It consists of a database schema that stores the output of various analysis tools as blobs, each of which the user can download as static views in a Web browser. GeneMine's focus on visual data mining makes it complementary to these useful bioinformatics databases.

The visual data-mining capabilities of GeneMine enable the scientist to rapidly explore the patterns and interconnections within the data that frequently suggest a functional hypothesis and provide a number of separate pieces of data for validating the hypothesis. Association rules can be revealed by annotation roll-up or drill-down using the sequence as an information hub. GeneMine shows icons next to each sequence representing the types of information it has discovered. Users can independently toggle each of these classes of data with a single click, giving them easy, intuitive access to the 2^N possible rollups of annotation data. For example, a user interested in structural features could select the predicted secondary structure (shown above the sequence in Figure 4) and directly validate by comparing with the homologous protein structure (by clicking on the annotation "Structure: Trypsin-like serine proteases"). The ability to combine information such as disease associations, genomic mapping, protein functional features, and structural conservation patterns (as in Figure 5 showing annotations for the human BRCA1 gene that reveals a likely DNA-binding domain [RING Zinc finger] corroborated by the pattern of conservation

Table 1 Example of how annotations integrate data from various sources

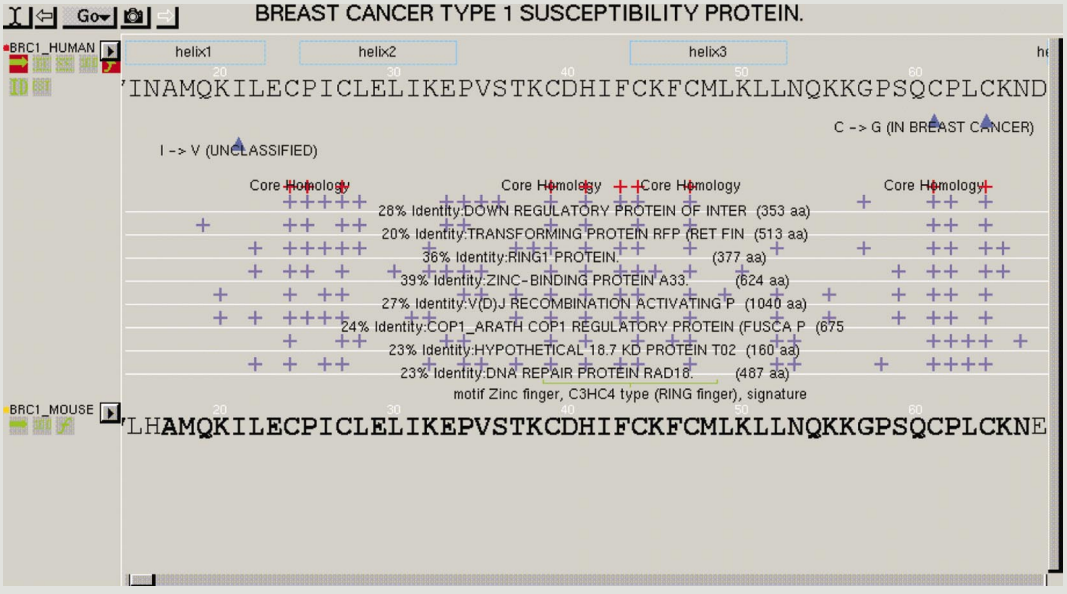
Protein functional motifs	PROSITE motif detection[26] Signal peptide prediction (PSORT)[31] Transmembrane regions (DAS, SOSUI, TMHMM)[32]
Secondary structure	From structure (PDB, SWISS-PROT, PIR) Predicted (PSSP, NNPredict) Coiled coils prediction (PBIL)
Structure/fold recognition	PDB BLAST2, ASTRAL, SCOP[27]
Protein domain detection	SWISS-PROT[30] BEAUTY, BLOCKS[33] ProDom[34] Protein linkage site prediction (DomCut)
Homology	Genbank, Entrez BLAST[4] PIR, NR, SWISS-PROT FASTA (IDEAS)
Gene expression patterns	DbEST[25] BLAST, tissue analysis
Protein features: active site residues, binding sites, post-translational modifications, disulfides	PIR, SWISS-PROT, Genbank
Protein family highly conserved residues	Performed on all homology analyses, novel method developed for GeneMine[2]
Gene prediction	ORF prediction (ALCES, NCBI) NR BLASTX protein homology detection
Polymorphisms	SWISS-PROT[30]
Drosophila homolog phenotypes	FlyBase[35]
Disease associations	Multiple sources
Molecular biology features	Restriction sites (Webcutter) Optimal PCR primer locations (ALCES) Repeat detection
Genomic mapping	STS matching, GeneMap99

across multiple protein families) provides a powerful "fast path" for identifying potential new drug targets and finding validating data (the primary application of GeneMine to date, in a large number of different pharmaceutical companies). Gene expression information (Figure 4) and phenotype information from model organisms (see Figures 3, 4, and 5) provide new connections to functional genomics strategies, all within the same integrating framework of GeneMine annotation.

Aggregation and clustering for multidimensional analysis

Data generalization and clustering techniques are essential in bioinformatics. The volume of results from bioinformatics analyses can be overwhelming, and the volume itself can obscure the important patterns in the data. The redundancy (repetitions of information that are really so similar as to be effectively the same) and noise (unreliable data) common in bioinformatics results encourages a "lazy" evaluation style in which scientists try to find a single "good hit" and ignore the rest of the "lower quality" information. However, this style deprives the scientist of many deeper patterns and unexpected insights that can be extracted from the mass of data. GeneMine seeks to automate this analysis through a combination of filtering (to remove noise), clustering (to reduce redundancy), and pattern analysis (to elucidate large-scale patterns within the total set of data). For example, in Figure 3 filtering of FASTA (a se-

Figure 5 Motif, homology, and conservation fingerprint annotations for the human BRCA1 gene



quence similarity search)[36] protein homologies yielded 110 significant hits (expectation score $<10^{-2}$), which were reduced to just five families via sequence clustering (with a 50 percent identity cutoff within each family), producing a view from which one can see the relevant functional groupings at a glance. (Figure 5 shows a similar reduction to just eight protein families.) For each of these families a phylogenetic analysis of what residues are most deeply conserved within that group yields a *conservation fingerprint* pattern, shown with + symbols (e.g., as in Figure 5). This pattern condenses an enormous amount of information from a full analysis of the homologs and their alignment, which would take quite a bit of effort for the user to assemble using a combination of other tools. It provides an immediately useful guide to key functional residues and a means for cross-validation of other information (discussed below).

Information in GeneMine is distilled through a series of levels. First, results are filtered for significance. An expectation score is attached to every result, and the user can freely adjust the threshold for showing or hiding results with lesser confidence levels. Second, homology results are clustered into families, typically producing an order of magnitude reduction. Third, the total set of annotations for each sequence are condensed by *type* via a set of clickable icons for each type, e.g., in Figure 3, secondary structure, homology, polymorphisms, three-dimensional (3-D) structure, identity match; conservation fingerprints; disease associations. Clicking each icon can either toggle the display of all annotations of that type for the sequence, or show a list from which the user can select. These annotation type icons show what kinds of information GeneMine has found, without taking up space with annotations the user currently does not wish to view.

Drill-down

GeneMine condenses information in its initial display, but preserves the links necessary for pursuing any piece of information that the user finds interesting. In Figure 3, for example, the user might be interested in learning more from the 3-D structure icon that appears next to the sequence name. Click-

ing this icon brings up annotations on the amino acid sequence marking regions of homology to known structures, e.g., "viral coat and capsid proteins." Clicking on one of these annotations brings up a list of all the homologous sequences detected in this family, in this case a list of seven sequences from PDB (Protein Data Bank) structures, with detailed statistics about their level and extent of homology, FASTA score, expectation score, etc. Another link shows in full detail the raw FASTA search results from which these hits were extracted. Each of the seven sequences in the list can also be clicked for a further drill-down menu with links to the structural classification of that protein (from the SCOP database[27]), information about its PDB structure, its ligands, etc., and a link to download this structure into GeneMine. Since the information window in GeneMine is a simple Web browser, these links can take advantage of any useful information sources on the Internet. Downloading the structure immediately displays its atomic coordinates and its amino acid sequence(s) in GeneMine, which will in turn launch a new cycle of automated alignment, analysis, and annotation of these new sequences. In short, each new piece of information leads seamlessly to many more sources of information; the user can easily browse these sources as deeply as he or she wants, expanding or collapsing the view of various kinds of information to explore different aspects of a problem without "losing the connection" to the other kinds of information that have been gathered.

Cross-validation

One of the major benefits of multiple, independent sources of information is the opportunity for cross-validation of predicted features and relationships. In the expert-query paradigm, the user tries a specific query in search of a specific kind of prediction; in GeneMine, many queries are performed against very different kinds of information sources, which are then assembled on the sequence as annotations. This method is very effective at making patterns of evidence *visible*, because all the data are projected onto the sequence alignment so that relevant information should be vertically aligned (to the sequence region responsible for the activity, be it an active site or entire domain) even if they come from separate sequences. For example, the accuracy of secondary structure prediction varies widely from protein class to class. In GeneMine one can easily check the accuracy of secondary structure predictions by turning on the auto-extend annotation mode that automatically aligns and annotates a representative set of medium-range homologs (clearly homologous but not very similar). Comparing the predicted secondary structures across the entire set of aligned sequences clearly reveals which features are robustly predicted across most of the members (despite their relatively dissimilar primary sequence), whereas other features are not consistently predicted across the set. Since the set is chosen to be similar enough to definitely share the same overall fold, the features that are predicted consistently across the set are very likely to be correct.

Figure 5 shows another example of cross-validation opportunities in GeneMine. Typically, BLAST users concentrate on the "best hit" in their homology report (BRC1_MOUSE, on the bottom line) and often pay scant attention to the dozens of low-homology hits (condensed by GeneMine into eight distinct homology families). Biologists are frequently uncomfortable with homologies in this 20–30 percent identity "twilight zone," wondering how to tell if they are real. Unfortunately, the best hit is often very uninformative because it is so similar to the query sequence that it adds little information. By contrast, distant homologies, if validated, can reveal deep functional patterns. In GeneMine one can see at a glance that the eight distant homology families appear to share common functional themes: gene regulation and DNA binding. An independent prediction from PROSITE[37] of a C3HC4 Zinc finger motif (below) at the center of this homology region corroborates this sharing. A quick drill-down view of information about this motif shows that its key feature is a pattern of conserved cysteines.[38] The conservation fingerprints of GeneMine for the eight protein families (shown as + signs marking the 10 percent most deeply conserved residues within each of these families) directly confirm this prediction by clearly highlighting three cysteines (positions 24, 27, and 39), a histidine (position 41), and four cysteines (positions 44, 47, 61, and 64) as the key functional residues in all eight families, and conserved in our sequence, even outside of the region identified by the PROSITE motif. Finally, mutation data for BRCA1 show that mutations at cysteines 61 and 64 actually lead to breast cancer,[39,40] confirming that they are essential for the normal function of this protein.

This is just one small example of the kinds of cross-validation that are possible when many different kinds and sources of information are integrated, aligned, and available for the scientist to explore easily and rapidly.

The visualization and modeling component of GeneMine (LOOK) was first developed by Lee in 1993 and has been used in many molecular modeling studies; for a small sample see References 41 and 42. The automated gene analysis and annotation system was developed in 1995. The software has been used extensively by commercial and academic researchers; for a small selection of recent papers see References 43 through 46. Subsequent to Lee's move to the University of California in January 2000, GeneMine was made freely available to academic researchers and is available for LINUX**, SGI, and Solaris** (see http://www.bioinformatics.ucla.edu/genemine for downloads or other information). The software may be used (via X-Windows**) on a PC or Macintosh** accessing a UNIX** server.

Acknowledgments

The authors wish to thank B. Modrek and A. Resch, who participated in mapping exons onto protein structure, and P. Thomas and R. Peccei for enabling the GeneMine software to be made freely available to academic researchers. The authors also wish to thank the many people who contributed to the evolution of the GeneMine software, including T. Tversky, P. Gentry, and A. Dalke, who wrote a number of sequence analysis and utility routines in the program; M. Mueller and T. Hatton, whose suggestions were always invaluable; C. Wang and H. Qin, who performed extensive testing of the software; and special thanks go to M. Levitt, whose SEGMOD homology modeling program works integrally with GeneMine.

Cited references

1. C. Lee, *LOOK: A Software System for Integrated Macromolecular Sequence—Structure Analysis and Modeling*, Molecular Applications Group, Palo Alto, CA (1993).
2. M. Mueller and C. Lee, *The GeneMine System for Automated Gene Function Analysis and Rich Structure-Function Annotation*, Molecular Applications Group, Palo Alto, CA (1995).
3. C. Marcazzo et al., "Identifying Gene Function and Features Through Comprehensive Automated Analysis," *Hilton Head Conference* (1997).
4. S. F. Altschul et al., "Basic Local Alignment Search Tool," *Journal of Molecular Biology* **215**, 403–410 (1990).
5. K. H. Buetow, M. N. Edmonson, and A. B. Cassidy, "Reliable Identification of Large Numbers of Candidate SNPs from Public EST Data," *Nature Genetics* **21**, 323–325 (1999).
6. K. Irizarry et al., "Genome-Wide Analysis of Single-Nucleotide Polymorphisms in Human Expressed Sequences," *Nature Genetics* **26**, 233–236 (2000).
7. R. Durbin and J. Thierry Mieg, *A C. elegans Database*, Medical Research Council, Cambridge, UK (1991).
8. V. M. Markowitz and O. Ritter, "Characterizing Heterogeneous Molecular Biology Database Systems," *Journal of Computational Biology* **2,** 547–556 (1995).
9. S. B. Davidson et al., "BioKleisli: A Digital Library for Biomedical Researchers," *Journal of Digital Libraries* **1**, 36–53 (1997).
10. L. C. J. Bailey et al., "GAIA: Framework Annotation of Genomic Sequence," *Genome Research* **8**, 234–250 (1998).
11. A. Kosky, E. Szeto, and V. M. Markowitz, *OPM Data Management Tools for CORBA Compliant Environments*, Lawrence Berkeley National Laboratories, Berkeley, CA (1996).
12. T. Gaasterland and C. W. Sensen, "MAGPIE: Automated Genome Interpretation," *Trends in Genetics* **12**, 76–78 (1996).
13. D. Frishman and H.-W. Mewes, "PEDANTic Genome Analysis," *Trends in Genetics* **13**, 415–416 (1997).
14. M. A. Andrade et al., "Automated Genome Sequence Analysis and Annotation," *Bioinformatics* **15**, 391–412 (1999).
15. V. Brusic and J. Zeleznikow, "Knowledge Discovery and Data Mining in Biological Databases," *Knowledge Engineering Review* **14**, 257–277 (1999).
16. M. S. Chen, J. Han, and P. S. Yu, *Data Mining: An Overview from Database Perspective*, IBM T. J. Watson Research Center, Yorktown Heights, NY (1996).
17. M. Hansen, D. Meads, and A. Pang, "Comparative Visualization of Structure-Sequence Alignments," *IEEE Conference on Information Visualization*, Research Triangle Park, NC (1998).
18. D. A. Payne et al., "OmniViz Pro: Applying Multiple Interactive Visualizations for the Life and Chemical Sciences," *IEEE Conference on Information Visualization*, Salt Lake City, UT (2000).
19. C. Lee and S. Subbiah, "Prediction of Protein Side-Chain Conformation by Packing Optimization," *Journal of Molecular Biology* **217**, 373–388 (1991).
20. M. Levitt, "Accurate Modeling of Protein Conformation by Automatic Segment Matching," *Journal of Molecular Biology* **226**, 507–533 (1992).
21. C. Lee, "Predicting Protein Mutant Energetics by Self-Consistent Ensemble Optimization," *Journal of Molecular Biology* **236**, 918–939 (1994).
22. Y. Wang et al., "Cn3D: Sequence and Structure Views for Entrez," *Trends in Biochemical Sciences* **25**, 300–302 (2000).
23. S. B. Needleman and C. D. Wunsch, "A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins," *Journal of Molecular Biology* **48**, 443–453 (1970).
24. T. F. Smith and M. S. Waterman, "Identification of Common Molecular Subsequences," *Journal of Molecular Biology* **147**, 195–197 (1981).
25. M. S. Boguski, T. M. Lowe, and C. M. Tolstoshev, "dbEST—Database for 'Expressed Sequence Tags'," *Nature Genetics* **4**, 332–333 (1993).
26. A. Bairoch, P. Bucher, and K. Hofmann, "The PROSITE Database, Its Status in 1995," *Nucleic Acids Research* **24**, 189–196 (1995).
27. A. G. Murzin et al., "SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures," *Journal of Molecular Biology* **247**, 536–540 (1995).
28. J. I. Garrels, "YPD—A Database for the Proteins of *Saccharomyces Cerevisiae*," *Nucleic Acids Research* **24**, 46–49 (1996).

29. C. A. Orengo et al., "CATH—A Hierarchic Classification of Protein Domain Structures," *Structure* **5**, 1093–1108 (1997).
30. A. Bairoch and R. Apweiler, "The SWISS-PROT Protein Sequence Data Bank and Its Supplement TrEMBL in 1998," *Nucleic Acids Research* **26**, 38–42 (1998).
31. K. Nakai and P. Horton, "PSORT: A Program for Detecting Sorting Signals in Proteins and Predicting Their Subcellular Localization," *Trends in Biochemical Sciences* **24**, 34–36 (1999).
32. E. L. Sonnhammer, G. von Heijne, and A. Krogh, "A Hidden Markov Model for Predicting Transmembrane Helices in Protein Sequences," *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology* (*ISMB*) **6**, 175–182 (1998).
33. S. Henikoff and J. G. Henikoff, "Protein Family Classification Based on Searching a Database of Blocks," *Genomics* **19**, 97–107 (1994).
34. F. Corpet et al., "ProDom and ProDom-CG: Tools for Protein Domain Analysis and Whole Genome Comparisons," *Nucleic Acids Research* **28**, 267–269 (2000).
35. W. M. Gelbart et al., "FlyBase: A *Drosophila* Database. The FlyBase Consortium," *Nucleic Acids Research* **25**, 63–66 (1997).
36. W. R. Pearson and D. J. Lipman, "Improved Tools for Biological Sequence Comparison," *Proceedings of the National Academy of Sciences (USA)* **85**, 2444–2448 (1988).
37. K. Hofmann et al., "The PROSITE Database, Its Status in 1999," *Nucleic Acids Research* **27**, 215–219 (1999).
38. K. L. B. Borden and P. S. Freemont, "The RING Finger Domain: A Recent Example of a Sequence-Structure Family," *Current Opinion in Structural Biology* **6**, 395–401 (1996).
39. L. H. Castilla et al., "Mutations in the BRCA1 Gene in Families with Early-Onset Breast and Ovarian Cancer," *Nature Genetics* **8**, 387–391 (1994).
40. L. S. Friedman et al., "Confirmation of BRCA1 by Analysis of Germline Mutations Linked to Breast and Ovarian Cancer in Ten Families," *Nature Genetics* **8**, 399–404 (1994).
41. M. Pantoliano et al., "Multivalent Ligand-Receptor Binding Interactions in the Fibroblast Growth Factor System Produce a Cooperative Growth Factor and Heparin Mechanism for Receptor Dimerization," *Biochemistry* 10229–10248 (1994).
42. P. Sengupta, H. A. Colbert, and C. L. Bargmann, "The *C. elegans* Gene odr-7 Encodes an Olifactory-Specific Member of the Nuclear Receptor Superfamily," *Cell* **79**, 971–980 (1994).
43. J. J. Chou et al., "Solution Structure of the RAIDD CARD and Model for CARD/CARD Interaction in Caspase-2 and Caspase-9 Recruitment," *Cell* **94**, 171–180 (1998).
44. C. Oxvig and T. A. Springer, "Experimental Support for a Beta-Propeller Domain in Integrin Alpha-Subunits and a Calcium Binding Site on Its Lower Surface," *Proceedings of the National Academy of Sciences (USA)* **95**, 4870–4875 (1998).
45. M. J. Shields et al., "The Effect of Human 2-Microglobulin on Major Histocompatibility Complex I Peptide Loading and the Engineering of a High Affinity Variant: Implications for Peptide-Based Vaccines," *Journal of Biological Chemistry* **273**, 28010–28018 (1998).
46. J. M. Goldberg and R. L. Baldwin, "A Specific Transition State for S-peptide Combining with Folded S-protein and Then Refolding," *Proceedings of the National Academy of Sciences (USA)* **96**, 2019–2024 (1999).

Accepted for publication January 10, 2001.

Christopher Lee *University of California, Los Angeles, Department of Chemistry and Biochemistry, Los Angeles, California 90095-1570 (electronic mail: leec@mbi.ucla.edu).* Dr. Lee is assistant professor and director of the UCLA bioinformatics program. He received a bachelor's degree in biochemistry and molecular biology from Harvard University in 1988, and a Ph.D. degree in structural biology from Stanford University in 1993. He has developed a variety of bioinformatics software, including CARA (protein mutant modeling), LOOK, and GeneMine (for gene function analysis and automated annotation), POA (multiple sequence alignment), and SNP-ASSESS (polymorphism detection). His recent research has focused on statistical analyses of human genome data, including identification of evidence for single nucleotide polymorphisms and alternative splicing. More information can be found at http://www.bioinformatics.ucla.edu/leelab.

Kris Irizarry *University of California, Los Angeles, Department of Chemistry and Biochemistry, Los Angeles, California 90095-1570 (electronic mail: irizarry@mbi.ucla.edu).* Mr. Irizarry is a Ph.D. degree candidate in the UCLA bioinformatics program and a research assistant in Dr. Lee's group. He has worked on a wide variety of research projects including *Drosophila* developmental genetics, identification of single nucleotide polymorphisms (SNPs), and construction of SNP-based maps of the human genome. Before coming to UCLA, he studied biochemistry and biophysics at Rensselaer Polytechnic Institute.