

Biological Applications of Multi-Relational Data Mining

David Page

Dept. of Biostatistics and Medical Informatics
and Dept. of Computer Sciences
University of Wisconsin
Madison, WI 53706
page@biostat.wisc.edu

Mark Craven

Dept. of Biostatistics and Medical Informatics
and Dept. of Computer Sciences
University of Wisconsin
Madison, WI 53706
craven@biostat.wisc.edu

ABSTRACT

Biological databases contain a wide variety of data types, often with rich relational structure. Consequently multi-relational data mining techniques frequently are applied to biological data. This paper presents several applications of multi-relational data mining to biological data, taking care to cover a broad range of multi-relational data mining techniques.

1. INTRODUCTION

Biological databases contain a wide variety of data. Consider storing in a database the operational details of a single-cell organism. At minimum one would need to encode the following.

- *Genome*: DNA sequence and gene locations
- *Proteome*: the organism's full complement of proteins, not necessarily a direct mapping from its genes
- *Metabolic pathways*: linked biochemical reactions involving multiple proteins, small molecules and protein-protein interactions
- *Regulatory pathways*: the mechanism by which the expression of some genes into proteins, such as transcription factors, influences the expression of other genes—includes protein-DNA interactions

In fact, such a database exists for part of what is known of the widely-studied model organism *E. coli*—EcoCyc [28]. This database contains other information as well besides the items listed above, such as operons (see Section 6).

Recording the diversity of data in the previous paragraph obviously requires a rich relational schema with multiple, interacting relational tables. In fact, even recording one type of data, such as metabolic pathways, requires multiple relational tables because of the graphical nature of pathways. And biological data can grow much more involved still. For example, when we move to multi-cellular organisms we need to include cell signaling and other aspects of one cell's interaction with other cells. Furthermore, because at present we do not have complete knowledge of the workings of any organism, biological databases often encode a variety of types of laboratory data that provide further insight into organism behavior. This may include data from the following high-throughput techniques:

- *Gene expression microarrays*: measure the degree to which each gene is being transcribed into messenger RNA (mRNA) under various conditions
- *Proteomics (primarily mass spectrometry)*: provides insight into which proteins are present in significant concentrations, as well as protein-protein interactions
- *Metabolomics*: measures the concentrations of small molecules (i.e., those with low molecular weight) in a sample
- *Single-nucleotide polymorphism (SNP) measurements*: SNP patterns are used as a surrogate for complete DNA sequences of multiple individuals, e.g. human patients—SNPs are single-base positions in DNA where individuals commonly differ

It is possible to gain insights from using any one of the data types described so far by itself. For example, standard machine learning and data mining algorithms have been applied to gene expression microarray data for a variety of purposes, as surveyed by Molla *et al.* [37]. Nevertheless, researchers in both data mining and biology are realizing that, often, stronger results can be obtained by using many of these data types together. Mining a database that contains several, or all, of these data types necessarily requires multi-relational techniques, either to analyze the data directly or to convert it into a single table in a useful way. The last two KDD Cup competitions have focused on biological databases and, not coincidentally, have highlighted the need for multi-relational data mining tools [9; 11].

The data types discussed so far, though many and diverse, only scratch the surface for biological databases. As researchers attempt to modulate the behavior of biological processes, for example through development of novel pharmaceuticals, still more types of data arise. Each year millions of compounds (molecules) are tested *in vitro* (in the “test tube”) and *in vivo* (in organisms) to see if they will bind to a target protein or have a desired biological effect. Even if a molecule has a desired effect, such as killing a harmful bacterium, it may be useless as a drug because it may be toxic to humans, it may break down too quickly or too slowly within the human body, it may not diffuse from the gut to the bloodstream (so it has to be taken by injection—unacceptable for some target activities), or it may not cross the blood-brain barrier (unacceptable, for example, for an anti-depressant or migraine medication). Therefore, molecules also are tested for these other factors,

giving rise to volumes of diverse data within pharmaceutical companies and university laboratories. Furthermore, the three-dimensional structures of molecules and target proteins, where known or estimated, are important items of information that require multiple relational tables to fully represent.

Both the amount and diversity of biological data will continue to grow rapidly because of a coming paradigm shift in biology, toward *systems biology*. As Hood and Galas (2003) note, whereas in the past biologists could study a “complex system only one gene or one protein at a time,” the “systems approach permits the study of all elements in a system in response to genetic (digital) or environmental perturbations.” Hood and Galas go on to state:

The study of cellular and organismal biology using the systems approach is at its very beginning. It will require integrated teams of scientists from across disciplines—biologists, chemists, computer scientists, engineers, mathematicians and physicists. New methods for acquiring and analyzing high-throughput biological data are needed [24].

In addition to data describing our current knowledge of organisms, novel types of high-throughput data, and data that arises from our attempts to alter organism behavior, another major type of biological data is semi-structured or text data. For example, the MEDLINE database [41] indexes more than 11 million articles in the biomedical literature, and contains abstracts and pointers to electronic versions for many of these articles. As another example, one of the web sites used most widely by biologists is GeneCards [46], which presents, for each human gene, information mined from biological text. Text data also has been integrated into the process of mining other types of data such as gene expression microarray data [53; 36].

Discussing in detail the variety of data types in biological databases is beyond the scope of any single paper. The goal of the present paper is to look more closely at a few examples of multi-relational data mining applied to biological data. The paper seeks to present a diversity of data types and of approaches, not focusing on any one particular method for multi-relational data mining.

2. BRIEF SURVEY OF BIOLOGICAL APPLICATIONS OF MULTI-RELATIONAL DATA MINING

Before focusing on a small number of biological applications of multi-relational data mining with which we are most familiar, we first present an overview of a variety of such applications. The earliest such work involved the application of inductive logic programming (ILP) to molecular data. Molecular data is a natural application for multi-relational data mining because no method has been found for representing molecules in ordinary feature-vector form without some loss of information.

In the first successful application of ILP to molecular data [29], the GOLEM program [39] was used to model the structure-activity relationships of trimethoprim analogues binding to dihydrofolate reductase.¹ The training data consisted of 44 trimethoprim analogues and their observed inhibition of *E.*

¹In structure-activity prediction, the task is to train a model

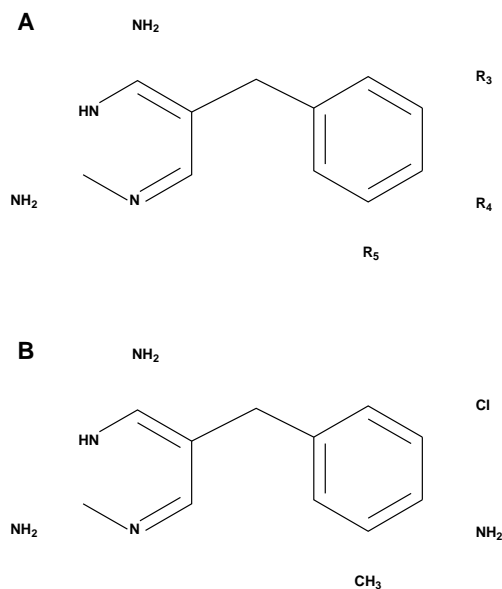


Figure 1: The family of analogues in the first ILP study. A) Template of 2,4-diamino-5(substituted-benzyl)pyrimidines: R₃, R₄, and R₅ are the three possible substitution positions. B) Example compound: R₃ — Cl; R₄ — NH₂; R₅ — CH₃

coli dihydrofolate reductase. Eleven additional compounds were used as unseen test data. GOLEM obtained rules that were statistically more accurate on the training data and on the test data than a previously published linear regression model. It was possible to run linear regression on this set because the molecules shared a common structure. Figure 1 shows the shared structure, or template, for all the molecules, as well as one particular instance of the template. Following shortly after this work [30] the two-dimensional atom-and-bond molecular descriptions of 229 aromatic and heteroaromatic nitro compounds [15] were given to the ILP system PROGOL [38]. Such compounds frequently are mutagenic. The study was confined to the problem of obtaining structural descriptions that discriminate molecules with positive mutagenicity from those which have zero or negative mutagenicity. A set of eight optimally compact rules were automatically discovered by PROGOL. These rules suggested three previously unknown features leading to mutagenicity. Attempts to force the multi-relational representation of the atom-and-bond structures into feature vectors resulted in the generation of examples containing millions of features each. Hence, unlike in the earlier molecular application, a claim can be made that this application truly needed multi-relational data mining. The mutagenicity data set remains a widely-used testbed for multi-relational learning algorithms. A number of interesting lines of research were motivated by this work on mutagenicity. First, this work led to a competition called the Predictive Toxicology Challenge 2000-2001 [22], in which both ordinary and multi-relational data mining tools were applied to the task of predicting rodent

to accurately predict which molecules have a particular biological activity (or to predict their degrees of the activity) from molecular structure. Details of the specific biological activities discussed in this section are beyond the scope of the paper.

carcinogenicity for 185 compounds slated to be tested in mice and rats. Further details and results can be found in the special issue of *Bioinformatics* (Vol. 19, No. 10) devoted to the challenge. Another line of research motivated by the previous work is *pharmacophore discovery*. This task requires data mining systems to move from the two-dimensional atom-and-bond representation of a molecule to a three-dimensional representation. Because a molecule may have multiple three-dimensional shapes, or *conformers*, this line of research has interesting links with multiple instance learning and is described in more detail in the next section. In another line of work, the Apriori-like ILP system WARMR was used to find all the molecular substructures (in the two-dimensional atom-and-bond structures) that appear with at least some minimum frequency in a set of molecules [16]. That work, which won a KDD Best Applied Paper Award, in turn motivated work on a molecular feature miner system, Molfea [32]. It also motivated work on Apriori-like algorithms for finding commonly-occurring subgraphs in a set of graphs [25; 57]. This is an exciting line of research that requires scaling Apriori's notion of frequent itemsets to apply to graphs, but doing so in such a way that the computational demands do not become overwhelming. One other piece of work in the area of mining molecular data was work on predicting the biodegradability of various compounds using ILP [18].

Another early application of multi-relational data mining was to protein secondary structure prediction [40]. We will not discuss this application because secondary structure prediction, while difficult, has proven amenable to feature-based approaches, such as neural networks as employed in the PHD system [50]. Nevertheless, protein tertiary structure, or three-dimensional topology, involves a variety of relationships among the portions of a protein and hence appears to demand the power of multi-relational data mining. Proteins are categorized into different “fold types,” or classes of three-dimensional structure. One notable application in this area is the use of ILP to classify proteins into different fold types [56].

Perhaps the most exciting new direction for biological applications of multi-relational data mining is in the construction of biological pathways from data. Section 4 goes into some detail about such an application of probabilistic relational models (PRMs), a multi-relational form of Bayesian networks. Other multi-relational techniques that have been applied to this task include graph learning [10] and ILP [5; 47]. One interesting twist in the application of ILP is the automatic proposal and execution of experiments. In this project, dubbed the “Robot Scientist Project,” the ILP system starts with a partial pathway model for biosynthesis of aromatic amino acids in yeast, expressed in logic. The system then formulates experiments that will provide it with information enabling it to complete its model, and a robot carries out the experiments. All experiments are auxotrophic growth experiments. In other words, they test whether some given yeast “knock-out”—yeast with one gene effectively removed—will grow on some particular medium, that is, in the presence of a certain set of nutrients. As the system obtains experimental results, it refines its model and, if necessary, performs further experiments.

Having presented an overview of biological applications of multi-relational data mining, we now turn to an more in-depth treatment of a small set of such applications. This

more in-depth treatment will permit us to better discuss lessons and challenges for such applications.

3. PHARMACOPHORE DISCOVERY

Motivated by the success of multi-relational data mining techniques to handle two-dimensional atom-and-bond molecular structure data, researchers in the pharmaceutical industry began to inquire whether these tools could be used to handle three-dimensional molecular structures. Such structures are of particular interest because the ability of a drug molecule to bind to its target protein typically is dependent on the ability of the drug molecule to fit into a pocket in the target protein and establish interactions based on charges or other properties of atoms. The 3D substructure of a molecule that interacts with the target is called a *pharmacophore*; identifying a pharmacophore can be an important step in the design of new drug molecules. For example, the following is a potential pharmacophore for ACE inhibition (a form of hypertension medication), where the spatial relationships are described through pairwise distances in units of Angstroms.² This pharmacophore was learned by an ILP system in an initial study applying multi-relational data mining to three-dimensional molecular structures. Figures 2 and 3 show two different ACE inhibitors with the parts of pharmacophore highlighted and labeled. The same ILP approach has been applied successfully to other related tasks and has been extended to predict activity levels [34].

```
Molecule A is an ACE inhibitor if
for some conformer Conf of A :
  molecule A contains a zinc binding site B;
  molecule A contains a hydrogen acceptor C;
  the distance between B and C in Conf is 7.9 +/- .75;
  molecule A contains a hydrogen acceptor D;
  the distance between B and D in Conf is 8.5 +/- .75;
  the distance between C and D in Conf is 2.1 +/- .75;
  molecule A contains a hydrogen acceptor E;
  the distance between B and E in Conf is 4.9 +/- .75;
  the distance between C and E in Conf is 3.1 +/- .75;
  the distance between D and E in Conf is 3.8 +/- .75.
```

Figure 4 illustrates how biomolecular activity and three-dimensional structure might be represented in a relational database. Attempts to represent the atom-and-bond structure and multiple conformations (three-dimensional shapes) of a molecule in a single file lead to loss of information for the data mining task. Even performing a natural join on the tables in Figure 4, while resulting in a single table, will not leave the features of the molecules aligned in a way that will permit standard feature-based algorithms to learn a pharmacophore. Correctly aligning the features resulting from the natural join would require almost complete knowledge of the target concept, or true pharmacophore.

Notice furthermore how the ILP-learned rule naturally addresses the multiple-instance problem [17]. While molecules (examples) are labeled as active or not, they obtain their activity through one or more of their conformers (instances). The rule labels a molecule *A* as active just if it has some conformer *Conf* that satisfies the rule.

²Hydrogen acceptors are atoms with a weak negative charge. Ordinarily, zinc-binding would be irrelevant; it is relevant here because ACE is one of several proteins in the body that typically contains an associated zinc ion. This is an automatically generated translation of an ILP-generated clause.

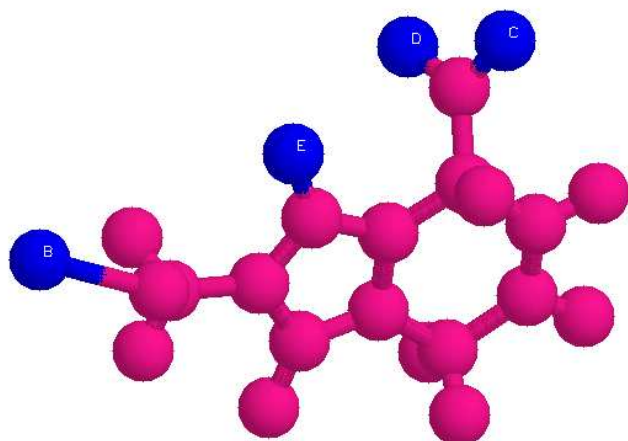


Figure 2: ACE inhibitor number 1 with highlighted 4-point pharmacophore.

Before closing this section, several areas of related work should be mentioned. First, the pharmaceutical industry has developed a variety of creative ways to encode molecules into feature vectors so that, even though some important information is lost, feature-based techniques can be applied from statistics, machine learning and data-mining. One example of such an encoding is the one used for Task 1 of KDD Cup 2001, predicting binding to thrombin [9]. Another example is the feature-generation stage of the COMPASS algorithm [26; 27].

One may conclude based on the preceding paragraph that multi-relational data mining algorithms are not always required in order to mine a multi-relational database. We take a broader view, that multi-relational techniques still are being employed in order to convert the task to a feature-based task. These multi-relational techniques may be domain-specific, as in the case of carefully-engineered feature-generation algorithms for molecular data, or they may be general-purpose. One example of a general-purpose algorithm is the ILP system RELAGGS, which was used to win Task 2 of KDD Cup 2001 [9]. RELAGGS was used to convert a multi-relational databases containing information about genes, gene expression, proteins, and protein-protein interactions into a single table. From this table, a model was learned using the support vector machine package SVM-light (svmlight.joachims.org) to predict protein function. Another general-purpose approach that has been employed to convert a multi-relational data set into a single table is the use of ILP to learn rules for a task, such as predicting mutagenicity; these ILP-learned rules are then used as features in order to produce a feature vector [55]. An example has a value of one for a given binary feature if and only if the ILP rule that corresponds to that feature applies to the example.

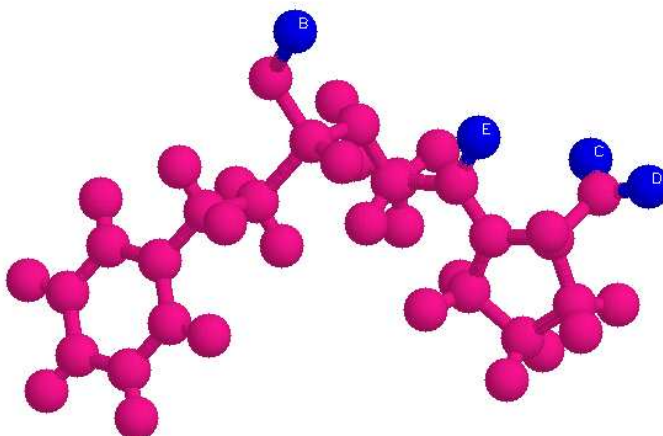


Figure 3: ACE inhibitor number 2 with highlighted 4-point pharmacophore.

Sternberg [56] using ILP to learn rules to predict the three-dimensional structural classes of proteins.

4. GENE REGULATION

As noted in the introduction, systems biology seeks to integrate data from a variety of related sources in order to construct whole-system models. One exciting example of multi-relational data mining in systems biology is the work of Segal *et al.* [52] in applying probabilistic relational models (PRMs) to induce gene regulatory networks from both sequence data and gene expression microarray data. Gene expression refers to the two-step process through which a gene is transcribed into mRNA, and that mRNA is translated into protein. Gene expression microarrays measure

Molecule	Target_1	...	Target_n
mol1	inactive		i inactive
mol2	active		inactive
.			
.			
.			

Molecular Bioactivity

Molecule	Bond_ID	Atom_1_ID	Atom_2_ID	Bond_Type
mol1	bond1	a1	a2	aromatic
.				
.				
.				

Bonds

Molecule	Conformer	Atom_ID	Atom_Type	X_Coordinate	Y_Coordinate	Z_Coordinate
mol1	conf1	a1	carbon	2.58	-1.23	0.69
.						
.						
.						

3D Atom Locations

Figure 4: Sample three-dimensional molecular database.

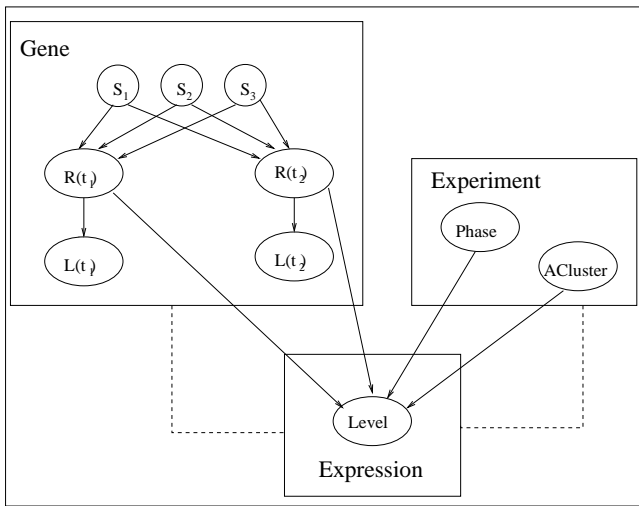


Figure 5: The PRM employed by Segal and colleagues. The *Expression* table records the measured mRNA level for each given $\langle Gene, Experiment \rangle$ pair. The *Phase* field of the *Experiment* table gives the experimental condition, while the *ACluster* field assigns experiments to clusters of similar experiments. The *Gene* relational table is simplified in the drawing; it actually contains nine nodes of the form $R(t_i)$ and nine nodes of the form $L(t_i)$, corresponding to nine known transcription factors. $R(t_i)$ tells whether transcription factor t_i actually can bind to a site just upstream of the given gene. $L(t_i)$ is a (noisy) measurement of whether such binding occurs. The S_j fields specify the bases in the DNA sequence upstream of the given gene; showing only three such fields is another simplification. In the database, the actual values for the $R(t_i)$ and $ACluster$ fields are not known; these are missing values, for which the learning algorithm will provide a prediction method.

the first part of this process, because mRNA levels are easier to measure than protein levels. In using microarray data, there often is a tacit assumption that transcription is a good surrogate for expression.

When the expression of one gene goes up, that gene's product (protein or mRNA) can influence the expression of other genes, or of that gene itself in a feedback loop. Such regulation is especially likely if the gene codes for a *transcription factor*. Transcription factors are proteins that bind to a subsequence of the DNA before a gene and encourage or repress the start of transcription. The subsequence to which a transcription factor binds is called the *transcription factor binding site*. If two genes have similar expression profiles, it is likely that they are controlled by the same transcription factor(s) and therefore have similar transcription factor binding sites in the sequence preceding them. Segal and colleagues seek to model both expression data, from microarrays, and sequence data.

A PRM can be viewed as a Bayes net where the variables of the Bayes net correspond to particular fields of a multi-relational database. The conditional probability distributions for a variable may be dependent on other variables from the same relational table or from other tables. In their application of PRMs to gene regulation, Segal and colleagues have a relational table for genes, a relational table

for experiments, and a relational table for gene expression microarray measurements, as shown in Figure 5. An expression measurement is for one gene within one experiment, i.e. under one set of experimental conditions. Therefore, in the database schema the expression measurement table captures the many-to-many relationship that exists between genes and experiments.

As with ordinary Bayes net learning given known structure, the task is to estimate the conditional probability distributions given data. This task is straightforward when there is no missing data. But in the present application, the values of the $R(t_i)$ fields (matching transcription factors with genes) and the $ACluster$ field (matching each experiment with a cluster of related experiments) are in fact unknown. Therefore an expectation-maximization (EM) algorithm is used to simultaneously learn which genes are controlled by which transcription factors and to cluster the experiments. The relationship between these two types of hidden variables is that under different experimental conditions, or in different clusters of experiments, different transcription factors will be active. While other work has been done on either modeling gene expression experiments or modeling the sequences of transcription factor binding sites, the novelty of this multi-relational application is that both types of modeling are done together, in synergy, using EM. The experimental results provide support for the claim of synergy, that modeling the two disparate types of data together is more successful than modeling either alone. We will return later to the general issue of predicting multiple aspects of gene regulation in concert.

5. INFORMATION EXTRACTION FROM TEXT

Many biological learning problems have a sequential nature. In this section and the next, we discuss two applications of this type and describe how they can be framed as multi-relational data mining problems. The first task that we consider is information extraction (IE) from the biological literature, which has received much attention recently [23]. In this task, one is interested in automatically recognizing and extracting specific classes of entities (e.g. proteins), relations among entities, or even complex events composed from relations. The ability to perform this task with high accuracy would be of great interest to many biologists. For example, this capability would significantly increase the productivity of the curators of genome databases who spend much of their time performing this task manually.

Figure 6 provides an illustration of the information extraction task. In this example, we are assuming that we are interested in identifying the names of proteins and the names of subcellular locations. Additionally, we are interested in extracting instances of the binary relation subcellular-localization, which represents the location of particular proteins within cells. We refer to an instance of a relation as a *tuple*. The top of the figure shows two sentences in an abstract, and the bottom of the figure shows the instances of the target classes and relation we would like to extract from the second sentence. The relation tuple asserts that the protein UBC6 is found in the subcellular compartment called the endoplasmic reticulum.

In order to learn models to perform this task, we could use training examples consisting of passages of text, annotated

“...
Here we report the identification of an integral membrane ubiquitin-conjugating enzyme. This enzyme, UBC6, localizes to the endoplasmic reticulum, with the catalytic domain facing the cytosol.
...”

↓

protein(UBC6)
location(endoplasmic reticulum)
subcellular-localization(UBC6,endoplasmic reticulum)

Figure 6: An example of the information extraction task. The top of the figure shows part of a document being processed. The bottom of the figure shows two extracted entities and one extracted relation instance.

with the entities and tuples that should be extracted from them. Since natural language has such rich structure, this task can naturally be framed as a multi-relational data mining problem.

Figure 7 shows a sentence representation that has been used in one biological IE project [45; 54]. This representation is based on syntactic parses produced by the Sundance [49] system. The sentence is segmented into typed phrases and each phrase is segmented into words typed with part-of-speech tags. For example, the second phrase segment is a noun phrase (NP_SEGMENT) that contains the protein name UBC6 (hence the PROTEIN label). In positive training examples, if a segment contains a word or words that belong to a domain in a target tuple, the segment and the words of interest are annotated with the corresponding domain. We refer to these annotations as *labels*. Test instances do not contain labels – the labels are to be predicted by the learned IE model.

Figure 7 illustrates some of the relationships that might be encoded in a representation for learning information-extraction models. There are others as well. Riloff [48] has developed semi-supervised IE methods that exploit subject-verb-object relationships. In learning models for recognizing protein names, Bunescu *et al.* [6] augment their representation with matches against a generalized dictionary of protein names. This generalized dictionary is created from a list of known protein names by replacing numbers with a generic number symbol $\langle n \rangle$, Roman letters with $\langle r \rangle$, and Greek letters with $\langle g \rangle$. For example, the protein name “NL-IL6-beta” would be generalized to “NF IL $\langle n \rangle$ $\langle g \rangle$.” Given a sentence to be processed, Bunescu *et al.* look for substrings that match an item in the generalized dictionary. Such matches are recorded in the input representation of the sentence. Some of these matches represent actual protein names, but others do not. The learner, however, can try to learn the conditions under which such matches do correspond to actual protein names, and it can learn to recognize protein names that do not involve dictionary matches.

A variety of learning algorithms have been used to induce models for information extraction from the biological literature. Bunescu *et al.* [6] have conducted an empirical investigation comparing a handful of learning algorithms and several hand-coded systems on two biological information-extraction tasks. In their experiments, the learned IE systems are consistently more accurate than the hand-coded

“This enzyme, UBC6, localizes to the endoplasmic reticulum, with the catalytic domain facing the cytosol.”

1	NP_SEGMENT	DET	this
		UNK	enzyme
2	NP_SEGMENT:PROTEIN	UNK:PROTEIN	ubc6
3	VP_SEGMENT	V	localizes
4	PP_SEGMENT	PREP	to
5	NP_SEGMENT:LOCATION	ART	the
		N:LOCATION	endoplasmic
		N:LOCATION	reticulum
6	PP_SEGMENT	PREP	with
7	NP_SEGMENT	ART	the
		N	catalytic
		UNK	domain
8	VP_SEGMENT	V	facing
9	NP_SEGMENT	ART	the
		N	cytosol
	(a)	(b)	(c)

Figure 7: Input representation for a sentence which contains a subcellular-localization tuple: the sentence is segmented into typed phrases and each phrase is segmented into words typed with part-of-speech tags. Phrase types and labels are shown in column (a). Word part-of-speech tags and labels are shown in column (b). The words of the sentence are shown in column (c). Note the grouping of words in phrases. The labels (PROTEIN, LOCATION) are present only in the training sentences.

ones. Among other algorithms, they evaluate several that employ relational representations, including RAPIER [8] and BWI [21]. In earlier work, Craven and Kumlien [12] used a relational learner [14] based on FOIL [43; 44] to learn information-extraction models for such problems. It is not clear yet which learning algorithms are best suited to biological IE tasks. However, a proposed series of community-wide evaluations [1] promises to shed more light on this issue.

From the early work in biological IE, one lesson that seems to hold is that rich, structured representations are beneficial. We consider one such case in more detail. The Craven group has developed an approach [45; 54] based on using *hierarchical* hidden Markov models [19] to extract information from the scientific literature. Hierarchical hidden Markov models have multiple “levels” of states which describe input sequences at different levels of granularity. This approach uses the input representation depicted in Figure 7, and thus each sentence is represented as a partially “flattened”, two-level description of each Sundance parse tree.

A schematic of one of their hierarchical HMMs is shown in Figure 8. The top of the figure shows the *positive model*, which is trained to represent sentences that contain instances of the target relation. The bottom of the figure shows the *null model*, which is trained to represent sentences that do not contain relation instances (e.g. off-topic sentences). At the “coarse” level, the hierarchical HMMs represent sentences as sequences of phrases. Thus, we can think of the top level as an HMM whose states emit phrases. This HMM is referred to as the *phrase HMM*, and its states are referred to as *phrase states*. At the “fine” level, each phrase is represented as a sequence of words. This is achieved by embedding an HMM within each phrase state. These embedded HMMs are referred to as *word HMMs* and their states as

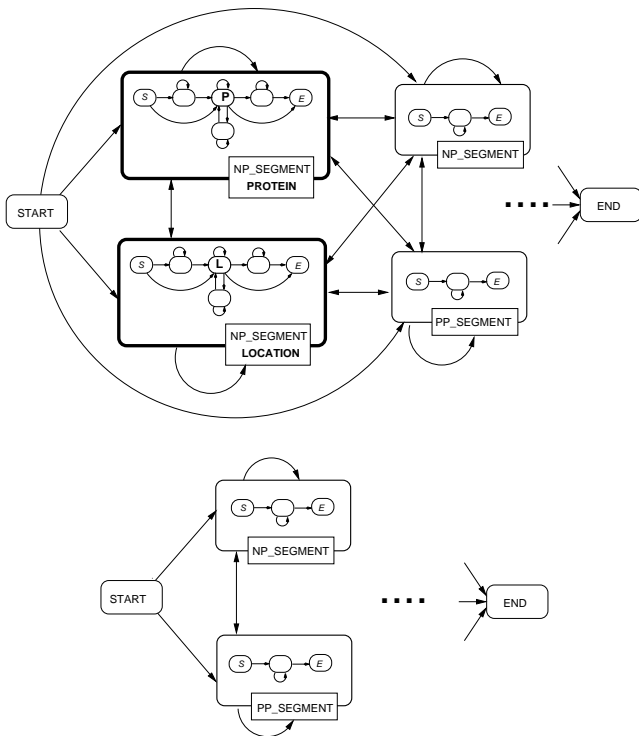


Figure 8: Schematic of the architecture of a hierarchical HMM for the subcellular-localization relation. The top part of the figure shows the positive model and the bottom part the null model. Phrase states are depicted as rounded rectangles and word states as ovals. The types and labels of the phrase states are shown within rectangles at the bottom right of each state. Labels are shown in bold and states associated with non-empty label sets are depicted with bold borders. The labels of word states are abbreviated for compactness.

word states. The phrase states in Figure 8 are depicted with rounded rectangles and word states are depicted with ovals. To explain a sentence, the HMM would first follow a transition from the START state to some phrase state q_i , then use the word HMM of q_i to emit the first phrase of the sentence, then transition to another phrase state q_j , emit another phrase using the word HMM of q_j and so on until it moves to the END state of the phrase HMM.

Like the phrases in the input representation, each phrase state in the HMM has a type and may have one or more labels. Each phrase state is constrained to emit only phrases whose type agrees with the state's type. Once a model has been trained, the Viterbi algorithm is used to predict tuples in test sentences. A tuple is extracted from a given sentence if the Viterbi path goes through states with labels for *all* the domains of the relation. For example, for the subcellular-localization relation, the Viterbi path for a sentence must pass through a state with the PROTEIN label and a state with the LOCATION label.

The experimental evaluation of this approach has shown that, in general, the more grammatical information incorporated into the models, the more accurate they are [54]. This result indicates the importance of using a multi-relational representation for the IE problem in biological domains.

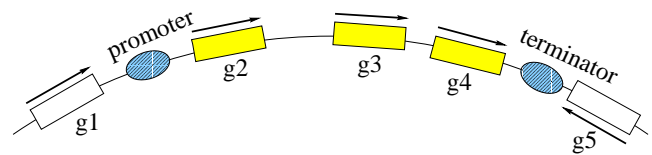


Figure 9: The concept of an operon. The curved line represents part of a bacterial chromosome and the rectangular boxes on it represent genes. An operon is a sequence of genes, such as $[g2, g3, g4]$ that is transcribed as a unit. Transcription is controlled via an upstream sequence, called a *promoter*, and a downstream sequence, called a *terminator*. A promoter enables the molecule performing transcription to bind to the DNA, and terminator signals the molecule to detach from the DNA. Each gene is transcribed in a particular direction, determined by which of the two strands it is located. The arrows in the figure indicate the direction of transcription for each gene.

6. SEQUENCE ANALYSIS

There are many data mining problems in molecular biology that involve representing and reasoning about DNA, RNA and protein sequences. Among these problems are gene finding [7], motif discovery [33], and protein structure prediction [51]. We argue that, increasingly, such sequence-analysis tasks are becoming multi-relational problems. The reasons for this trend are twofold. First, other sources of evidence, in addition to the sequences themselves are being leveraged in the learning and inference processes. Second, many sequence elements of interest are composed from other sequence elements. In this section, we discuss these issues in more detail, focusing in particular on our recent work in analyzing bacterial genomes. [3; 4; 13].

Recall that the first step in the process of a gene being expressed is for it to be *transcribed* into a similar RNA sequence. Although the expression of a gene can be regulated at various points, the most significant regulatory controls are exerted on the transcription process. For example, a gene can be “shut off” by preventing it from being transcribed. In some organisms, especially bacteria, there are certain sets of contiguous genes, called *operons* that are transcribed coordinately. In other words, the genes in an operon are “turned on” or “shut off” as a unit.

Figure 9 illustrates the concept of an operon. The transcription process is initiated when a molecule called *RNA polymerase* binds to the DNA before the first gene in an operon. The RNA polymerase binds to a special sequence called a *promoter*. It then moves along the DNA using it as a template to produce an RNA molecule. When the RNA polymerase gets past the last gene in the operon, it encounters a special sequence called a *terminator* that signals it to release the DNA and ceases transcription.

Consider the task of analyzing and annotating a newly sequenced genome. More than 100 bacterial genomes have been sequenced [2], and several hundred others are in the process of being sequenced, so this is an increasingly common task to be addressed. In addition to identifying the genes in the genome and predicting something about their functions, we would also like to identify regulatory elements such as operons, promoters, terminators, and others. Often, it is the case that we know about some experimentally determined instances of these elements.

Figure 10 shows a sample database that represents what

OperonID	Name	GeneID	Name	Function	LeftCoord	RightCoord	Operon
Op1	aceBAK	b4014	aceB	enzyme	4213057	4214658	aceBAK
Op2	alkA	b4014	aceB	enzyme	4214688	4215992	aceBAK
Op3	araBAD	b4047	yjbL	?	4257900	4258154	?
•		•					
•		•					
•		•					

Operons		Genes				
ProbeID	Coordinate	Exp1	Exp2	Exp3	Exp4	Exp5
p1	16646	-2.10	0.14	3.01	-1.23	0.69
p2	16663	-1.91	0.21	3.01	-1.05	0.82
•						
•						
•						

Promoters		Terminators	
PromID	Coordinate	TermID	Coordinate
Prom1	4212991	Term1	4216005
Prom2	421980	Term2	4257870
•		•	
•		•	
•		•	

Figure 10: A sample database describing known attributes of a bacterial genome.

might be known about a particular bacterial genome. The operon table lists known operons. The gene table describes attributes of known genes including their sequence coordinates and which operon they are contained within, if this is known. The promoter and terminator tables list a reference sequence coordinate for each known instance. The expression table contains expression measurements from microarrays, across a variety of experimental conditions. The table also lists a reference sequence coordinate for each *probe* that is used to take an expression measurement. Note that the sequence coordinates provide a means for computing other relations among entities, such as the linear order of known genes, promoters and terminators in the genome.

Given a data set such as that shown in Figure 10, there are a number of prediction tasks that we might want to address. These include predicting additional genes, promoters and terminators, and predicting how genes are organized into operons, in cases where this is not known. Note that these tasks are clearly multi-relational in that the representation includes attributes of genes (e.g. sequence coordinates), relations among genes (e.g. containment in the same operon), and relations between genes and other entities, such as promoters and terminators. Additionally, the records in the expression table are linked to these entities via their sequence coordinates.

In our research, we have considered various approaches to recognizing regulatory elements in bacterial genomes. In our most recent work [4], we have developed a probabilistic language model to simultaneously predict promoters, terminators and operons. This model is, for the most part, a hidden Markov model, although two components of it are actually stochastic context free grammars. Most of the relational structure in this problem is sequential, and thus probabilistic language models are especially suited to the task.

We argue that those interested in relational data mining should consider probabilistic language models, such as HMMs, to be powerful tools in their arsenal. For many key biological sequence-analysis tasks, such as gene finding [7], these methods are the ones that provide state-of-the-art predictive accuracy.

One exciting topic of research in this area is investigating ways in which learned models can take into account all of the relevant data that is available. For example, there are several new gene-finding HMM methods [31; 35] that take sequences from multiple organisms as input, and use the evolutionary relationships among the organisms to aid in gene prediction. As another example, our bacterial genome model takes gene expression data (as illustrated in Figure 10) as input in addition to sequence data. In both of these cases, the augmented representation has resulted in improved predictive accuracy. We conjecture that this trend of leveraging additional data will lead to representations that are even more multi-relational than current ones.

7. LESSONS AND CHALLENGES

One interesting lesson that biological applications such as pharmacophore discovery have provided is how naturally multi-relational data mining address the multiple instance problem. This has been noted earlier, for example in the original pharmacophore discovery paper [20]. Recently, Perlich and Provost have provided a useful hierarchy of data mining tasks, based on representation, where multiple instance tasks form a class subsumed by multi-relational tasks [42].

A second lesson, important though perhaps not novel, is that multiple distinct sources of information can provide improved accuracy. For example, the application of PRMs used both expression data and sequence data to accurately recognize which genes are controlled by the same transcription factors. As another example, two consecutive genes are likely to be in the same operon if their expression patterns are correlated, if they are separated by a relatively small number of bases, and if no promoters or terminators appear between them; using these multiple sources of information provides improved operon prediction over using any one source alone. Multi-relational data mining is naturally suited to using distinct sources of information that can be encoded in distinct relational tables.

One major challenge that biological applications highlight for the area of multi-relational data mining is simply computational complexity. Biological databases are growing at an exponential rate. Moreover, the time to simply check a potential hypothesis or discovery against the data can be much higher for a multi-relational system than for an ordinary single-table or feature-vector system. For example, the task of determining whether a molecule contains a given substructure is the NP-complete task of subgraph isomorphism.

A second, related challenge that biological applications raise for multi-relational data mining systems is ease of use. A biologist can run a decision-tree learner or a linear regression routine on his or her data with relatively little training and preprocessing. But running a multi-relational data mining system typically requires a much larger knowledge engineering effort. Research is needed into ways of making multi-relational data mining systems easier to use by biologists or

other domain experts.

8. CONCLUSION

Multi-relational data mining tools have been applied to a variety of biological tasks. Nevertheless, the breadth of inter-related data types discussed in the introduction is significantly greater than in any of the specific applications we discussed. Furthermore, with the growth in systems biology, the breadth and relational structure of biological databases will only increase. Therefore, biological databases provide a major challenge for multi-relational data mining. Already, data mining researchers should be more ambitious in applying multi-relational algorithms to larger and more diverse databases. For example, we should be integrating data on DNA (SNPs), mRNA (from gene expression microarrays), proteins (from mass spectrometry) and metabolomics into multi-relational analyses. We should be drawing on relevant text sources as well. Stretching ourselves in this manner in applications is sure to reveal new research directions in the development of multi-relational data mining algorithms.

9. ACKNOWLEDGEMENTS

M.C. was supported by NSF grant IIS-0093016 and NIH grant R01-LM07050-01. D.P. was supported by NSF grant 9987841.

10. REFERENCES

- [1] Critical assessment of information extraction systems in biology, 2003. www.pdg.cnb.uam.es/BioLink/BioCreative.eval.html.
- [2] A. Bernal, U. Ear, and N. Kyrpides. Genomes On-Line database (GOLD): A monitor of genome projects worldwide. *Nucleic Acids Research*, 29(1):126–127, 2001.
- [3] J. Bockhorst, M. Craven, D. Page, J. Shavlik, and J. Glasner. A Bayesian network approach to operon prediction. *Bioinformatics*, 19(10):1227–1235, 2003.
- [4] J. Bockhorst, Y. Qiu, J. Glasner, M. Liu, F. Blattner, and M. Craven. Predicting bacterial transcription units using sequence and expression data. *Bioinformatics*, 19(Suppl. 1):34–43, 2003.
- [5] C. Bryant, S. Muggleton, S. Oliver, D. Kell, P. Reiser, and R. King. Combining inductive logic programming, active learning, and robotics to discover the function of genes. *Electronic Transactions in Artificial Intelligence*, 2001.
- [6] R. Bunescu, R. Ge, R. Kate, R. Mooney, E. Marcotte, and A. Ramani. Learning information extractors for proteins and their interactions. In *Working Notes of the ICML Workshop on Machine Learning in Bioinformatics*, 2003.
- [7] C. Burge and S. Karlin. Prediction of complete gene structures in human genomic DNA. *Journal of Molecular Biology*, 268:78–94, 1997.
- [8] M. E. Califf and R. Mooney. Bottom-up relational learning of pattern matching rules for information extraction. *Journal of Machine Learning Research*, 4:177–210, 2003.
- [9] J. Cheng, C. Hatzis, H. Hayashi, M.-A. Krogel, S. Morishita, D. Page, and J. Sese. KDD Cup 2001 report. *SIGKDD Explorations*, 3(2):47–64, 2002.
- [10] L. Chrisman, P. Langley, S. Bay, and A. Pohorille. Incorporating biological knowledge into evaluation of causal regulatory hypotheses. In *Proceedings of the Eighth Pacific Symposium on Biocomputing*. 2003.
- [11] M. Craven. The genomics of a signaling pathway: A KDD cup challenge task. *SIGKDD Explorations*, 4(2):97–98, 2003.
- [12] M. Craven and J. Kumlien. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, pages 77–86, Heidelberg, Germany, 1999. AAAI Press.
- [13] M. Craven, D. Page, J. Shavlik, J. Bockhorst, and J. Glasner. A probabilistic learning approach to whole-genome operon prediction. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 116–127, La Jolla, CA, 2000. AAAI Press.
- [14] M. Craven and S. Slattery. Relational learning with statistical predicate invention: Better models for hyper-text. *Machine Learning*, 43(1-2):97–119, 2001.
- [15] A. Debnath, R. L. de Compadre, G. Debnath, A. Schusterman, and C. Hansch. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of Medicinal Chemistry*, 34(2):786 – 797, 1991.
- [16] L. Dehaspe, H. Toivonen, and R. King. Finding frequent substructures in chemical compounds. In R. Agrawal, P. Stolorz, and G. Piatetsky-Shapiro, editors, *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98)*. AAAI Press, New York, 1998.
- [17] T. Dietterich, R. Lathrop, and T. Lozano-Perez. Solving the multiple-instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71, 1997.
- [18] S. Džeroski, H. Blockeel, B. Kompare, S. Kramer, B. Pfahringer, and W. V. Laer. Experiments in predicting biodegradability. In *Proceedings of the Ninth International Workshop on Inductive Logic Programming*, pages 80–91. Springer-Verlag LNAI 1634, 1999.
- [19] S. Fine, Y. Singer, and N. Tishby. The hierarchical hidden Markov model: Analysis and applications. *Machine Learning*, 32:41–62, 1998.
- [20] P. Finn, S. Muggleton, D. Page, and A. Srinivasan. Discovery of pharmacophores using Inductive Logic Programming. *Machine Learning*, 30:241–270, 1998.

- [21] D. Freitag and N. Kushmerick. Boosted wrapper induction. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, pages 577–583, Austin, TX, 2000. AAAI Press.
- [22] C. Helma and S. Kramer. A survey of the predictive toxicology challenge 2000-2001. *Bioinformatics*, 19(10):1179–1182, 2003.
- [23] L. Hirschman, J. Park, J. Tsujii, L. Wong, and C. Wu. Accomplishments and challenges in literature data mining for biology. *Bioinformatics*, 18:1553–1561, 2002.
- [24] L. Hood and D. Galas. The digital code of DNA. *Nature*, 421:444–448, 2003.
- [25] A. Inokuchi, T. Washio, and H. Motoda. An apriori-based algorithm for mining frequent substructures from graph data. In *Principles of Data Mining and Knowledge Discovery*, pages 13–23, 2000.
- [26] A. Jain, T. Dietterich, R. Lathrop, D. Chapman, R. Critchlow, B. Bauer, T. Webster, and T. Lozano-Pérez. Compass: A shape-based machine learning tool for drug design. *Journal of Computer-Aided Molecular Design*, 8:635–652, 1994.
- [27] A. Jain, K. Koile, B. Bauer, and D. Chapman. Compass: Predicting biological activities from molecular surface properties. *Journal of Medicinal Chemistry*, 37:2315–2327, 1994.
- [28] P. Karp, M. Riley, S. Paley, and A. Pellegrini-Toole. EcoCyc: Electronic encyclopedia of E. coli genes and metabolism. *Nucleic Acids Research*, 25(1), 1997.
- [29] R. King, S. Muggleton, R. Lewis, and M. Sternberg. Drug design by machine learning: The use of inductive logic programming to model the structure-activity relationships of trimethoprim analogues binding to dihydrofolate reductase. *Proceedings of the National Academy of Sciences*, 89(23):11322–11326, 1992.
- [30] R. King, S. Muggleton, A. Srinivasan, and M. Sternberg. Structure-activity relationships derived by machine learning: the use of atoms and their bond connectives to predict mutagenicity by inductive logic programming. *Proceedings of the National Academy of Sciences*, 93:438–442, 1996.
- [31] I. Korf, P. Flicek, D. Duan, and M. Brent. Integrating genomic homology into gene structure prediction. *Bioinformatics*, 17(Suppl. 1):S140–S148, 2001.
- [32] S. Kramer, L. D. Raedt, and C. Helma. Molecular feature mining in HIV data. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-01)*, pages 136–143, 2001.
- [33] C. Lawrence, S. Altschul, M. Boguski, J. Liu, A. Neuwald, and J. Wootton. Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. *Science*, 262:208–214, 1993.
- [34] N. Marchand-Geneste, K. Watson, B. Alsberg, and R. King. A new approach to pharmacophore mapping and qsar analysis using inductive logic programming. application to thermolysin inhibitors and glycogen phosphorylase b inhibitors. *Journal of Medicinal Chemistry*, 45(2):399–409, January 2002.
- [35] I. Meyer and R. Durbin. Comparative ab initio prediction of gene structures using pair HMMs. *Bioinformatics*, 18(10):1309–1318, 2002.
- [36] M. Molla, P. Andrae, J. Glasner, F. Blattner, and J. Shavlik. Interpreting microarray expression data using text annotating the genes. *Information Sciences*, 146:75–88, 2002.
- [37] M. Molla, M. Waddell, D. Page, and J. Shavlik. Using machine learning to design and interpret gene-expression microarrays. *AI Magazine*, 2003. (In Press).
- [38] S. Muggleton. Inverse entailment and Progol. *New Generation Computing*, 13:245–286, 1995.
- [39] S. Muggleton and C. Feng. Efficient induction of logic programs. In *Proceedings of the First Conference on Algorithmic Learning Theory*, Tokyo, 1990. Ohmsha.
- [40] S. Muggleton, R. King, and M. Sternberg. Protein secondary structure prediction using logic-based machine learning. *Protein Engineering*, 5(7):647–657, 1992.
- [41] National Library of Medicine. Pubmed, 1999. <http://www.ncbi.nlm.nih.gov/PubMed/>.
- [42] C. Perlich and F. Provost. Aggregation-based feature invention and relational concept classes. In *Proceedings of KDD-03*. ACM SIGKDD, August 2003.
- [43] J. R. Quinlan. Learning logical definitions from relations. *Machine Learning*, 5:239–266, 1990.
- [44] J. R. Quinlan and R. M. Cameron-Jones. FOIL: A midterm report. In *Proceedings of the European Conference on Machine Learning*, pages 3–20, Vienna, Austria, 1993.
- [45] S. Ray and M. Craven. Representing sentence structure in hidden Markov models for information extraction. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 1273–1279, Seattle, WA, 2001. Morgan Kaufmann.
- [46] M. Rebhan, V. Chalifa-Caspi, J. Prilusky, and D. Lancet. Genecards: Encyclopedia for genes, proteins and diseases, 1997. <http://bighost.area.ba.cnr.it/GeneCards>.
- [47] P. Reiser, R. King, D. Kell, S. Muggleton, C. Bryant, and S. Oliver. Developing a logical model of yeast metabolism. *Electronic Transactions in Artificial Intelligence*, 2001.
- [48] E. Riloff. An empirical study of automated dictionary construction for information extraction in three domains. *Artificial Intelligence*, 85:101–134, 1996.
- [49] E. Riloff. The sundance sentence analyzer, 1998. <http://www.cs.utah.edu/projects/nlp/>.

- [50] B. Rost and C. Sander. Combining evolutionary information and neural networks to predict protein secondary structure. *Proteins*, 19:55–77, 1994.
- [51] S. Schmidler, J. Liu, and D. Brutlag. Bayesian segmentation of protein secondary structure. *Journal of Computational Biology*, 7:233–248, 2000.
- [52] E. Segal, B. Taskar, A. Gasch, N. Friedman, and D. Koller. Rich probabilistic models for gene expression. *Bioinformatics*, 1:1–10, 2001.
- [53] H. Shatkay, S. Edwards, W. J. Wilbur, and M. Boguski. Genes, themes and microarrays: Using information retrieval for large-scale gene analysis. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 317–328, La Jolla, CA, 2000. AAAI Press.
- [54] M. Skounakis, M. Craven, and S. Ray. Hierarchical hidden Markov models for information extraction. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, Acapulco, Mexico, 2003. Morgan Kaufmann.
- [55] A. Srinivasan and R. King. Feature construction with inductive logic programming: A study of quantitative predictions of biological activity aided by structural attributes. In S. Muggleton, editor, *Proceedings of the 6th International Workshop on Inductive Logic Programming*, pages 352–367. Stockholm University, Royal Institute of Technology, 1996.
- [56] M. Turcotte, S. Muggleton, and M. Sternberg. Automated discovery of structural signatures of protein fold and function. *Journal of Molecular Biology*, 306:591–605, 2001.
- [57] X. Yan and J. Han. Closegraph: Mining closed frequent graph patterns. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'03)*, 2003.