Exploring Life Sciences Data Sources

Zoé Lacroix
Arizona State University
zoe.lacroix@asu.edu

Felix Naumann
Humboldt University of Berlin
naumann@informatik.hu-berlin.de

Louiqa Raschid
University of Maryland
louiqa@umiacs.umd.edu

Maria Esther Vidal
Simon Bolivar University
mvidal@ldc.usb.ve

1 Introduction

There has been an explosion of the data that is available to the biomolecular researcher. A recent estimate suggests up to 600 public data sources! While this explosion presents an opportunity, it is accompanied by difficulties in harnessing and exploring this data. An average research group can (simultaneously) utilize up to 40 databases many of which are publicly available on the Web. Public life science data sources represent a complex link-driven federation of sources. A fundamental problem facing the researcher today is correctly identifying a specific instance of a biological entity, e.g., a specific gene or protein, and then obtaining a complete functional characterization of this entity instance by exploring a multiplicity of inter-related and inter-linked sources. An example question that could be answered by such a correct and complete characterization is as follows: What cancer-related proteins have been identified and what relevant knowledge has been collected by other researchers over the past two years?

Life science data sources contain data on classes of scientific entities such as genes and sequences. Each source may have data on one or more classes. There is significant diversity in the *coverage* of these sources. For example, NCBI Nucleotide, DDBJ and EMBL Nucleotide have different attributes characterizing (describing) sequences, but they all cover the same sequences. On the other hand, while AllGenes, RatMap and the Mouse Genome Data base (MGD) all contain data on genes, they are targeted at different organisms. MGD covers mouse genes and RatMap covers rat genes. However, AllGenes contains both human and mouse genes, so there is an overlap between AllGenes and MGD. See [Nau02] for a detailed discussion on various database related aspects of coverage.

Relationships between scientific objects are often implemented as physical links between data sources. Each physical link between sources may be visualized as a collection of individual links, going from a data object in one source to another data object, in the same or a different source. The physical implementation of these links may vary, e.g., embedded identifiers, `URL`s, etc. Properties of the relationship such as uni or bi-directional, 1:1 or 1:N, etc. may also vary widely.

A scientist is often interested in exploring relationships between scientific objects, e.g., genes and citations. These objects may be retrieved from various data sources, e.g., PubMed for publications. Such an exploration process typically starts from one or more of these available sources, and continues by following direct links, e.g., a URL, or traversing paths, i.e., concatenations of links via intermediate sources.

Given some start class in source S and target class in source T, there may be multiple alternate paths. Each path potentially yields very different results with different *properties*. This depends on the following: the attributes characterizing each source; the intermediate sources and corresponding entity classes that are traversed in a path; and the contents of each source and each physical link between sources. An example property is *result cardinality*, i.e., the number of data objects of the target class T that are obtained by starting from (a relevant set of) data objects in S. Note that result cardinality may vary based on the choice of the path.

These properties are of interest from a number of perspectives. For example, from a query evaluation viewpoint, one can predict the cost of evaluating a query given some specific sources and paths. This can impact query optimization. One could also choose specific sources and paths depending on some criteria that are evaluated on these properties, e.g., to maximize result cardinality or to maximize the number of attributes. Such criteria impact the domain

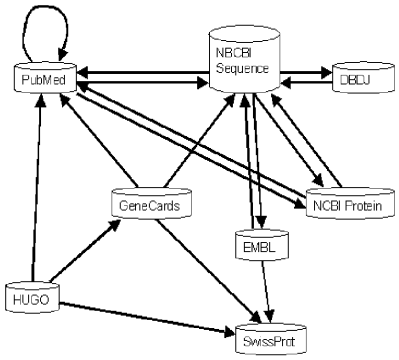

Figure 1: A physical graph (PG) of Life Science Sources

specific semantics of the results.

There has been prior research on providing access to life science sources [EKL00, ELR01, KRG99, TKM99]. Example systems include DiscoveryLink [HKR+00], Kleisli and its successors [DCB+01, Won00], SRS [EA93, EV97] and Tambis [PSB+99]. Typically, these systems have not explored the semantics of alternate sources and overlap in content. Recent research in [MHTH01, MSHTH02] has addressed this problem but they do not consider the semantics of multiple alternate paths. Properties of links have been studied in the context of XML document processing [PG02a, PG02b]; they do not consider semantics associated with paths.

In this paper, we summarize two research tasks. The first task involves algorithms to explore the search space of links and paths between biological data sources, and to efficiently identify paths that are relevant to a query expressed by a scientist [LRV03]. The second task is to develop a framework to determine the properties that characterize (multiple alternate) links and paths between two sources [LNR03]. Together, these tasks provide a solid foundation to support scientific exploration.

2 Physical and Logical Map of Sources

We model life science data sources at two levels: the logical and physical level. The physical level consists of the actual data sources and the links that exist between them. An example of data sources and links is in Figure 1. The physical level is modeled by a directed graph PG. Nodes represent data sources. Edges represent a physical implementation of a link between two data sources. A data instance in one data source may have a link to one or more data instances in the other data source, e.g., a gene in GeneCards links to a citation in PubMed. The semantics of this physical link may vary, e.g., is it bi-directional? how many objects participate? is it 1:1 or 1:N? A physical path in PG is defined in a straightforward manner by traversing the links of PG.

The logical level consists of entity classes, i.e., concepts or ontology classes. Entity classes are implemented by one or more physical data sources or possibly parts of data sources. For example, a logical entity class Citation may be implemented by the data source PubMed. An entity class Sequence may be implemented by NCBI Nucleotide, DDBJ and EMBL. Each source that implements an entity class will provide a unique identifier for each entity instance and will include attribute values that characterize the instance. There may be as many identifiers for an entity instance as there are sources. In some cases, there may be multiple identifiers for *equivalent* instances in the same source. Instances are semistructured with respect to all the attribute values defined by that source.

ENTITY	DATA SOURCE
Sequence (seq)	NCBI Nucleotide EMBL DDBJ
Protein (prt)	NCBI Protein SwissProt
Citation (cit)	NCBI PubMed

Table 1: A Possible Mapping from Entity Classes to Physical Data Sources

An example of a possible mapping from entity classes to data sources in PG is in Table 1. We note that the identification of entity classes and the mapping from these classes to physical sources may not be unique. However, one can consider that a typical or commonly accepted mapping exists.

3 Exploring Paths Between Sources

To allow a scientist to explore relationships or associations between the logical entities, we consider a simple language based on regular expressions. While we recognize that such a language has limited expressive power and cannot express branching, predicates, etc. it is sufficient for us to illustrate how queries are answered on data sources. Given a regular expres-

sion as input, the objective is to interpret the regular expression on the graph PG.

We refer to the set of entity classes as E, and use the notation p: protein; s: sequence; g: gene; c: citation. For our example, we consider data sources Nucleotide, EMBL, DDBJ, Protein, SwissProt, HUGO, GeneCards and PubMed.

Query 1: p.c

Result: Protein → PubMed, SwissProt → PubMed

Query 1 expresses a query to retrieve all citations linked to proteins. Entity p can be interpreted by either the NCBI Protein data source or SwissProt (Table 1. Entity c can be interpreted by PubMed. There is a link from NCBI Protein to PubMed, and there is a link from SwissProt to PubMed (Figure 1. Therefore, both links Protein → PubMed and SwissProt → PubMed are possible interpretations of the regular expression. We note that these links have different physical inplementations. The link Protein → PubMed may correspond to the Entrez capability to search for a Citation from a Protein object. SwissProt → PubMed may correspond to hyperlinks embedded in the presentation of proteins in SwissProt.

Query 2: g.*e*.c

Note that the symbol e represents any entity in the set of entity classes E.

Entity g may be interpreted by both GeneCards and HUGO. e may be interpreted by any data source. From Figure 1, there are 3 outgoing links from GeneCards to PubMed, Sequence and SwissProt, respectively. Similarly, there are 3 outgoing links from HUGO to GeneCards, PubMed and SwissProt, respectively. There is a link from PubMed to PubMed; thus, GeneCards → PubMed → PubMed and HUGO → PubMed → PubMed are paths that match the regular expression. There is link from Sequence to PubMed; thus, GeneCards → Sequence → PubMed is a solution. There is a link from SwissProt to PubMed; thus, GeneCards → SwissProt → PubMed and HUGO → SwissProt → PubMed are solutions. Finally, there is a link from GeneCards to PubMed; thus, HUGO → GeneCards → PubMed is a solution. To summarize, this query has 6 possible interpretations, including self references (loops) on PubMed.

A similar problem has been addressed in [MW89] where it was shown that for (any) graph and regular expression, determining if a particular edge occurred in a path that satisfied the regular expression and was in the answer was NP hard and that finding all paths was NP complete.

Our research addresses the following tasks:

- Efficient algorithms based on automata to explore the search space of life sciences sources. The solution is polynomial in the size of the PG when the PG is acyclic.
- Developing semantics to fully explore sources and to choose paths that reflect user's needs.
- Use of heuristics to limit the exploration of the search space and to efficiently identify paths that satisfy the regular expression.

We briefly review some examples of semantics.

- Path length semantics – [Minimize | Maximize] the number of different sources that are visited in a path assuming that PG is such that a node may be visited more than once.
- Attribute cardinality semantics – [Minimize | Maximize] the (total) number of attributes of all the entities visited along the path.
- Result cardinality semantics – [Maximize | Minimize] the cardinality of the results.

See [LRV03] for details on the algorithm to identify paths that satisfy a regular expression and experiments on the use of heuristics to efficiently identify paths that satisfy the regular expression and the semantics.

4 Characterizing Properties of Paths

Consider a start source S and end source T. There are two tasks that are of interest. The first task is estimating (predicting) properties such as the result cardinality, for any path from S to T. The second is comparing the properties of alternate paths. Comparing alternate paths could compare their relative result cardinalities. Alternately, they could determine the overlap in the data objects in T, for each path. Finally, they could obtain a (partial) order of paths based on either the result cardinality or overlap.

A physical graph can have several roots and leaves. This occurs since there may be multiple sources representing the root entity class, and/or several sources for the leaf entity class. For simplicity, we assume a graph with unique root and unique leaf, and one source per entity class. Thus, the logical and physical graphs are the same.

Figure 2 shows a physical graph with four sources (node); each source is annotated with the scientific entity class. Each link represents physical links between the sources. We note that these 4 sources are NIH/NCBI sources.

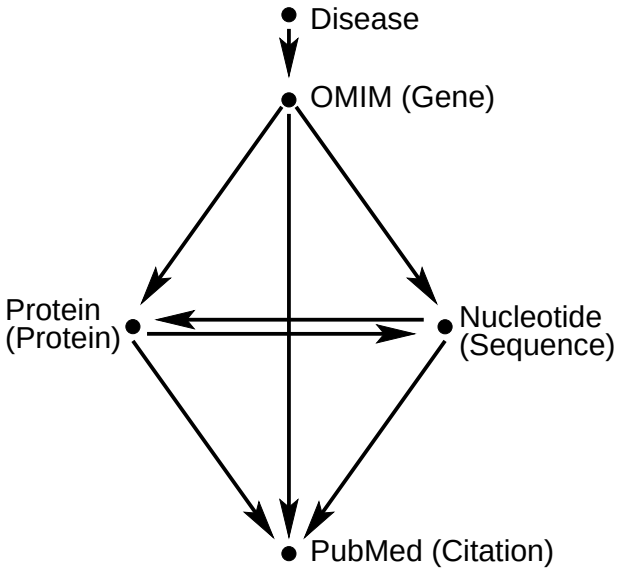

Figure 2: The physical graph PG_1 with sources (and scientific entities)

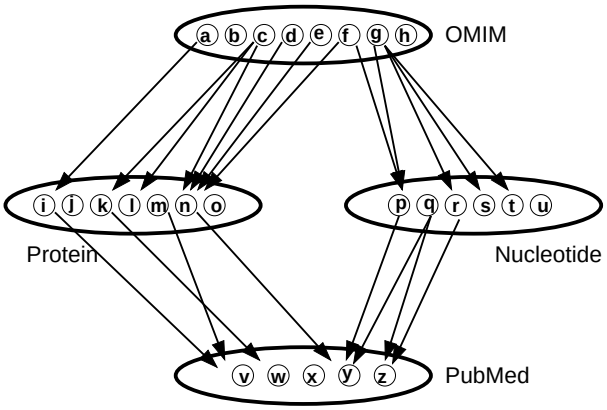

Figure 3: The data graph DG_1

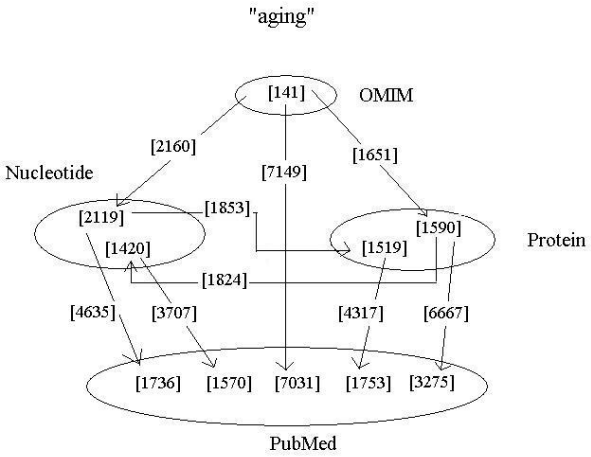

Figure 4: A result graph RG_1

Figure 3 shows a sample data graph DG_1. Informally, each source in the data graph stores a set of objects (labeled `a`, `b`, `c`, etc.) and a set of links (unlabeled).

Finally, a result graph RG_1 as shown in Figure 4, is a subset of the data graph. Details on how the data was collected is in [LNR03]; this graph is circa February 2003. In this example, the result graph starts with OMIM data objects that are related to the medical condition `aging` and terminates in citations from PubMed. There are 5 alternate paths `om-pu`, `om-nu-pu`, `om-nu-pr-pu`, `om-pr-pu` and `om-pr-nu-pu`, where `om`, `nu`, `pr` and `pu` represent the 4 sources, OMIM, Nucleotide, Protein and PubMed, respectively. For ease of presentation, we do not display all links from one data object to another. Instead, for each link between two sources, S_i and S_{i+1}, we note the number of data objects in S_i that have links to S_{i+1}, and the number of links from S_i to S_{i+1}. Note that there may be duplicate objects in S_{i+1}. for example, in Figure 4, 141 OMIM objects had links to 7149 PubMed objects; however, there were only 7031 unique PubMed objects in this set[1].

In [LNR03], we develop a framework to estimate result cardinalities of paths. The framework is flexible and can exploit available statistics on DG. We note that for the NIH/NCBI data sources, accurate up-to-date statistics on DG_1 is maintained [LL03]. However, most data providers typically may not maintain accurate statistics. We validate the accuracy of this framework for predicting the shape and result cardi-

[1]The notion of duplicate objects in PubMed is defined with respect to the internal identification of the source, e.g., MEDLINE identifiers for citations in PubMed.

nality for several results graphs that are a subset of DG_1. Our research addresses the following issues:

- Develop a framework to use statistics obtained from the DG to completely characterize the properties of a link and a path. Typical properties include *Participation*, *Image*, and *Outdegree*. Informally, given a link from S_i to S_{i+1}, the *Participation* is the number of objects of S_i that participate, and the *Image* is the number of objects of S_{i+1} that participate, in the link.
- Refine the framework to consider dependencies between links, e.g., if there is a link from an object in S_i to an object in S_{i+1}, then the probability that there is a link from the object in S_{i+1} to an object in S_{i+2} *may be different* from the probability that any object in S_{i+1} has a link to an object in S_{i+2}.
- Compare overlap between alternate paths. Use overlap and / or result cardinality to obtain a (partial) order of paths.

As an example, starting from 141 OMIM records in RG_1, the direct link from OMIM to PubMed yielded the largest number (7149) PubMed records; of these 7031 were unique. Paths of length 2, through Nucleotide or Protein, yielded fewer records, and paths of length 3 yielded even fewer records. Note that there may be duplicates among the records. For example 141 OMIM records yielded 2160 Nucoleotides (2119 unique Nucleotide) records, and 4635 PubMed records (1736 unique PubMed records) via these Nucleotide records.

Next, we consider overlap between paths. Of the 7149 PubMed records (7031 unique) in the direct link from OMIM to PubMed, 661 records were in overlap with the 4635 records (1736 unique records) obtained from the link from OMIM to PubMed via Nucleotide. Of these same 7149 records (7031 unique), 941 were in overlap with the 6667 records (3275 unique) obtained from OMIM to PubMed via Protein. Finally, of the 4635 records (1736 unique) obtained from the link from OMIM to PubMed via Nucleotide and the 6667 records (3275 unique) obtained from OMIM to PubMed via Protein, 1531 records were in overlap.

Due to space limitations, we are unable to present our analysis and results. We refer the reader to [LNR03] for details on the framework and results on accuracy in predicting result cardinality.

5 Conclusions

In this paper, we reviewed the challenges of exploring life sciences sources, where multiple sources describe scientific entity classes and there are multiple alternate links between sources. We then reviewed two research tasks: The first task explores the search space of links and paths between biological data sources, and efficiently identifies paths that are relevant to a query expressed by a scientist. The second task is to develop a framework to determine the properties that characterize (multiple alternate) links and paths between two sources. Together, these tasks provide a solid foundation to support scientific exploration.

6 Acknowledgements

We thank David Lipman and Alex Lash of NIH/NCBI for their expertise on NCBI data sources and for providing statistics, Barbara Eckman of IBM Life Sciences for discussions on life sciences exploration, and Damayanti Gupta and Hyma Murthy for data collection and analysis.

References

[DCB+01] S. Davidson, J. Cabtree, B. Brunk, J.Schug, V. Tannen, C. Overton, and C. Stoeckert. K2/kleisli and gus: Experiments in integrated access to genomic data sources. *IBM Systems Journal*, 40(2), 2001.

[EA93] T. Etzold and P. Argos. Srs: An indexing and retrieval tool for flat file data libraries. *Computer Applications of Biosciences*, 9(1), 1993.

[EKL00] B. Eckman, A. Kosky, and L. Laroco. Extending traditional query-based integration approaches for functional characterization of post-genomic data. *BioInformatics*, 17(2), 2000.

[ELR01] B. Eckman, Z. Lacroix, and L. Raschid. Optimized seamless integration of biomolecular data. *Proceedings of the IEEE International Symposium on Bio-Informatics and Biomedical Engineering*, 2001.

[EV97] T. Etzold and G. Verde. Using views for retrieving data from extremely heterogeneous databanks. *Pacific Symposium on Biocomputing*, pages 134–141, 1997.

[HKR+00] L. Haas, P. Kodali, J. Rice, P. Schwarz, and W. Swope. Integrating life sciences data - with a little garlic. *Proceedings*

of the IEEE International Symposium on Bio-Informatics and Biomedical Engineering, 2000.

[KRG99] G. Kemp, C. Robertson, and P. Gray. Efficient access to biological databases using corba. *CCP11 Newsletter*, 3.1(7), 1999.

[LL03] A. Lash and D. Lipman. Statistics on nih/ncbi data sources. *Personal communication*, 2003.

[LNR03] Z. Lacroix, F. Naumann, and L. Raschid. Characterizing properties of paths in biological data sources. *In preparation*, 2003.

[LRV03] Z. Lacroix, L. Raschid, and M.E. Vidal. Exploring the search space of paths in biological data sources. *In preparation*, 2003.

[MHTH01] P. Mork, A. Halevy, and P. Tarczy-Hornoch. A model for data integration systems of biomedical data applied to online genetic databases. *Proceedings of the AMIA*, 2001.

[MSHTH02] P. Mork, R. Shaker, A. Halevy, and P. Tarczy-Hornoch. Pql: A declarative query language over dynamic biological data. *Proceedings of the AMIA*, 2002.

[MW89] Alberto O. Mendelzon and Peter T. Wood. Finding regular simple paths in graph databases. In *Proceedings of the International Conference on Very Large Data Bases*, pages 185–193, 1989.

[Nau02] Felix Naumann. *Quality-driven Query Answering for Integrated Information Systems*, volume 2261 of *Lecture Notes on Computer Science (LNCS)*. Springer Verlag, Heidelberg, 2002.

[PG02a] N. Polyzotis and M. Garofalakis. Statistical synopses for graph-structured xml databases. *Proceedings of the ACM SIGMOD Conference*, 2002.

[PG02b] N. Polyzotis and M. Garofalakis. Structure and value synopses for xml data graphs. *Proceedings of the Very Large Data Base Conference*, 2002.

[PSB+99] N.W. Paton, R. Stevens, P.G. Baker, C.A. Goble, S. Bechhofer, and Brass. Query processing in the tambis bioinformatics source integration system. *Proceedings of the IEEE Intl. Conf. on Scientific and Statistical Databases (SSDBM)*, 1999.

[TKM99] T. Topaloglou, A. Kosky, and V. Markovitz. Seamless integration of biological applications within a database framework. *Proceedings of the Intl. Conf. on Intelligent Systems for Molecular Biology (ISMB)*, 1999.

[Won00] L. Wong. Kleisli: Its exchange format, supporting tools, and an application protein interaction extraction. *Proceedings of the IEEE International Symposium on Bio-Informatics and Biomedical Engineering*, 2000.