A Graph Framework for Multimodal Medical Information Processing

Georgios Drakopoulos

MDAKM Lab
University of Patras
Patras 26504 Hellas
Email: drakop@ceid.upatras.gr

Vasileios Megalooikonomou

MDAKM Lab
University of Patras
Patras 26504 Hellas
Email: vasilis@ceid.upatras.gr

***Abstract*—Multimodal medical information processing is currently the epicenter of intense interdisciplinary research, as proper data fusion may lead to more accurate diagnoses. Moreover, multimodality may disambiguate cases of co-morbidity. This paper presents a framework for retrieving, analyzing, and storing medical information as a multilayer graph, an abstract format suitable for data fusion and further processing. At the same time, this paper addresses the need for reliable medical information through co-author graph ranking. A use case pertaining to frailty based on Python and Neo4j serves as an illustration of the proposed framework.**

Keywords–Frailty index; Co-morbidity; Neo4j; Tensor analysis; Multimodal data mining

I. INTRODUCTION

Multimodal medical information processing is a technique for providing improved diagnosis by fusing data from heterogenous sources. This is especially useful when clinical data are scarce or difficult to obtain. Moreover, multimodal processing may provide a solution to co-morbidity cases, namely in cases where individuals suffer from at least two diseases with overlapping symptoms. As a rule, co-morbidity cases are hard to treat since all symptoms can be mistakenly attributed to only a single disease, complicating thus the cure. For instance, a specific brain anomaly may be attributed by medical practitioners to lesions based on EEG readings but functional neuroimaging may reveal additional brain damage. Medical information processing systems are currently expected to handle a multitude of data forms including, among others, research papers, field reports with raw data, medical images, EEG waveforms, and MEG recordings.

In order to support multimodality, a versatile and generic data structure should be used. The framework proposed in this work relies extensively on multilayer graphs, namely graphs whose labeled edges belong to at least two distinct categories. With the advent of large scale graph processing systems such as Apache Giraph, of graph oriented machine learning tools such as GraphLab, and of graph databases such as Neo4j and GraphDB, there is a plethora of high quality, open source, software tools using graphs as the default storage and query format to select from.

The primary contribution of this work is Perseus. The name is a direct reference to the mythological Hellenic hero Perseus. His main achievement was the direct handling of Medusa, a creature whose hair were deadly snakes. In a very abstract representation, snakes can be drawn as simple edges connecting two vertices. Also, in a liberal interpretation, their venom corresponds to an exceedingly degree of complexity. Perseus is a conceptual framework for retrieving, maintaining, querying, and processing medical data from a number of sources. Perseus is strongly based on the explicit assumption that data as well as their interconnection patterns can be expressed as graphs. To demonstrate the potential of Perseus to be tailored to specific application needs, a use case for the frailty index, a significant health indicator for the elderly, is presented.

The remainder of this paper is organized as follows. In section II the scientific literature regarding multimodal medical information processing is overviewed. Subsequently, in section III Perseus is presented and analyzed. The frailty index use case of section IV illustrates the application of multilayer graphs to medical cases with potential co-morbidity. Finally, in section V future research directions are outlined.

II. PREVIOUS WORK

Fully multimodal information processing systems are a relatively new breakthrough. However, there have been notable recent efforts towards a true multimodal system. In [17] document similarity functions are merged with document ranking. Text mining information from medical documents based on the grammatical structure as well as on field reports and possibly text metadata is outlined in [1] and in [3].

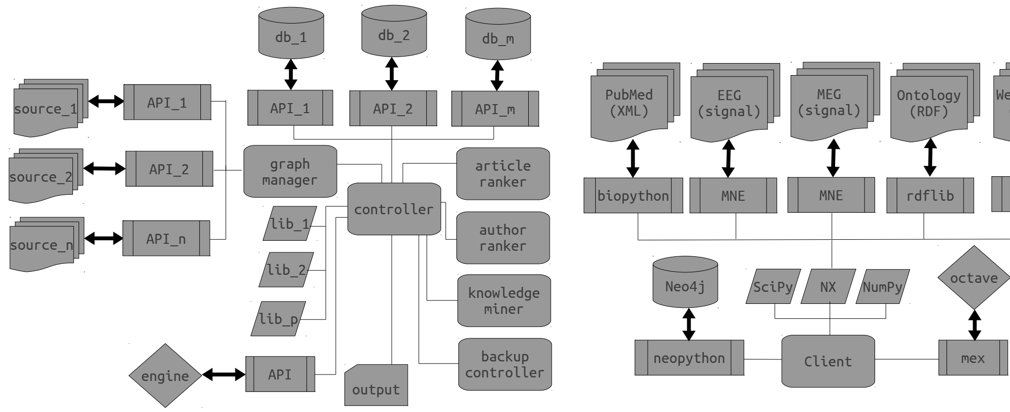
The advent of Semantic Web sparked additional interest in multimodality in gene ontology [5][4] as well as in reasoning wether specific genes are related to a given disease [2][14]. Further momentum was gained with the addition of retrieval methods which are based on medical images [19][17], which allowed more targeted and context aware database search. To the best of the authors knowledge, no medical multimodal data mining system currently exists that relies on a graph database at the back end, although [16] describes a graph database implementation for human gene ontology processing.

III. FRAMEWORK

A. Architecture

In figure 1(a) the Perseus architecture is displayed. It is centered around the contoller module, which directs data acquistion from a number of heterogenous sources through the appropriate interfaces. Data collected can be stored either in their original form or after processing in various databases depending on the operational needs.

Multimodality is ensured by obtaining the appropriate APIs for at least two different types of data sources. Besides



(a) Perseus generic architecture

(b) Frailty index specific architecture

locally available clinical results and articles from open access databases such as PubMed, common sources include medical ontologies, statistical aggregations from organizations like WHO, and biosignal repositories.

Notice that the star architecture, which has a sigle critical point of failure, combined with the nature of a medical processing system calls for a backup controller which is normally shadowing the main controller and replacing it, should the need arise.

Versatile storage is ensured through multilayer graphs. They have been selected as the primary data structure not only because many significant data sources provide data in a tree or graph format, such as PubMed, but also it facilitates information fusion in a space efficient manner [13]. Additionally, graphs allow the easy information localization as well as the easy composition of a summary based on various sources. Finally, the connectivity patterns of an author-to-author collaboration network and of a document-to-document citation network allow the ranking of both the authors and the articles, serving thus as a quality control mechanism.

IV. Use Case: Frailty Index

A. Architecture

As a concete example of its true multimodal nature, Perseus can be tailored to integrate and maintain information about the frailty index, a major health indicator for the elderly. More formally

Definition 1: Frailty is a biological syndrome of decreased reserve and resistance to stressors resulting from cumulative declines across multiple physiological systems and causing vulnerability to adverse outcomes.

With the sharp ageing of population globally, frailty index has risen in significance as a factor of social coherence. Moreover, proper elderly monitoring through frailty index can prevent an array of fatal accidents.

The tailored architecture is illustrated in figure 1(b). It integrates information from a wide ragne of sources, namely field reports from Web forms, EEG and MEG biosignals, PubMed articles, and medical ontologies which have been explicitly developed for frailty. Notice that data source selection is indicative, as an alternative Perseus implementation might as well include sensor readings from mobile devices or smart homes or results from Rorsach tests. Notice that besides the frailty index, its confidence level can be determined from factors listed in table III.

Algorithm 1 shows the information flow for the frailty index case.

Algorithm 1 Frailty index information flow

Require: Quality criteria list T
Ensure: Frailty index I is computed
Ensure: Confidence level L is computed
Collect data from Web forms through NLTK
Extract a set of frailty related terms $\{\tau_k\}$
for all terms τ_k through Biopython **do**
  Retrieve PubMed documents $\{d_i\}$ containing τ_k
  if $\{d_i\}$ meets the criteria of T **then**
    Retrive PubMed popularity scores of $\{d_i\}$
  end if
end for
Retrieve EEG and MEG waveforms through MNE
while unexplained symptoms remain **do**
  Retrieve a frailty ontology O through rdflib
  if O matches symptoms **then**
    Store O in Neo4j
  end if
end while
return computed (I, L)

Python has been selected as the primary language because of its ability to generate glue code, namely code connecting two or more software components whithout modifying them. Moreover, there are Python interfaces for each of the other tools.

B. Biopython and Entrez

Entrez is a special purpose, federated search engine designed to provide seamless access across a vast array of medical databases. Currently Entrez APIs have been developed for various programming languages and problem solving environments, most notably the Matlab API, the BioGo package for Go, and the Java Entrez Eclipse API.

Entrez stores PubMed documents as XML trees, facilitating thus lexical analysis. At the same time it renders appealing the use of a graph database for local storage and analysis. Table

TABLE I. PUBMED XML DOCUMENT TAGS.

FileHeader	ArticleSet	Article
Journal	PublisherName	JournalTitle
Issn	Volume	Issue
PubDate	Year	Month
Season	Day	Replaces
ArticleTitle	VernacularTitle	FirstPage
ELocationID	Language	AuthorList
Author	FirstName	MiddleName
LastName	Suffix	CollectiveName
Affiliation	Identifier	GroupList
Group	GroupName	IndividualName
PublicationType	ArticleIdList	ArticleId
History	Abstract	OtherAbstract
CopyrightInfo	ObjectList	Object
Param	LastPage	

I lists the mandatory and optional tags from the Entrez XML schema.

Each API for Entrez should implement as a minimum functionality methods for searching medical documents based either on ID or on search terms, retrieving document abstracts and bodies, and retrieving relevant articles. Usually methods for text parsing are provided by a number of APIs.

Biopython is a Python open source Entrez API. The most common Biopython methods are listed in table II. Furthermore, Python is heavily employed in the NLP field, mainly because of NLTK, the natural language toolkit. Finally, even if certain Perseus components are written in another language, Python is strongly famed for its so called glue code, namely code that provides connection and communication between different applications or modules of the same application, offering thus a unified view of an otherwise segmented software.

TABLE II. BIOPYTHON FUNCTIONS.

Method	Description
efetch	Retrieves records from a list of ids
epost	Posts a file containing a list of ids in the user environment for future use
esearch	Searches and retrieves ids and term translations and optionally retains results
elink	Checks for the existence of an external or Related Articles link in a list of ids
einfo	Provides field index term counts, last update, and available links for each database
esummary	Retrieves document summaries
egquery	Provides Entrez database counts in XML for a single search using Global Query
espell	Retrieves spelling suggestions
read	Parses the XML results returned by any of the above functions
parse	Parses the XML results returned by functions. Appropriate for large files.

C. Neopython and Neo4j

Since the beginning of the current decade there has been a rising interest in four branches of non-relational databases collectively referred to as NoSQL databases. For a brief NoSQL review see [15].

Neo4j is a major graph database supporting queries to large graphs through either Cypher, an ASCII art high level query language, or through APIs for various programming languages, most notably Python, Scala, and Java. It is also possible to query graphs in Neo4j in SPARQL through suitable extensions. Neo4j is currently open source and it is written mostly in Scala.

Property 1: Neo4j is schemaless.

Property 2: The three operational Neo4j requirements are basic availability, soft state, and eventual consistency collectively known as BASE [15].

Compared to relational ACID requirements, the BASE ones are easier to implement and place less strain on the database system, including fewer data duplications and locks. The downside is that the system does not become immediately consistent.

Property 3: The primary conceptual data model supported by Neo4j and offered to a high level user through Cypher is the property graph [15].

For a review of the graph property model see [7][6].

Neopython is a Python interface for Neo4j which allows the formation of dynamic Cypher queries as Python strings. For this reason, Neopython has been included to the proposed implementation.

D. Document Ranking Criteria

> *But when a rule is extremely complex, what it is in conformity with it passes for irregular.*
> (Leibniz)

Building T requires care, as it can easily degrade to a nearly all-reject rule. In fact, as a safety valve in real life situations there should be a component evaluating the effectiveness of T by checking the Perseus null return rate. Significant criteria for T are listed in table III.

TABLE III. DOCUMENT RANKING CRITERIA.

Article	
PubMed popularity	age
citation graph ranking	number of authors
journal impact factor	acceptance rate
Content	
terminology frequency	keywords
attached data	reproducible research
Authors	
co-authorship graphs	research continuity

The ranking of the article depends on at least three group of factors. First, the article is ranking as a member of a document collection. This can be achieved by examining any scientific citation graph. Then, the actual article contents contribute to article ranking. Finally, the authors are ranked, for instance from any established research collaboration graph, and their reputation affects article ranking. Research continuity means that the authors should have written other articles on the same subject. Reproducible research means that the data are available along with the publication or in a public Web location, giving thus the opportunity to other research teams to process them.

E. Graph Analytics

Perseus relies heavily on graph analytics in order to perform computations. Specifically Perseus

- ranks authors in co-authorship graphs.
- ranks articles in citation graphs.
- parses PubMed XML documents.
- converts EEG and MEG waveforms as graphs.
- reasons on frailty ontologies.
- derives brain regions of interest.

Graph algorithms for performing the above tasks are listed in table IV.

TABLE IV. HIGHER ORDER GRAPH ANALYTICS.

Structure	
Density	Expander graph
Bridges	Articulations
Triangles	Squares
Euler path	Hamilton path
Connected components	Independent sets
Total connectivity	Cheeger number
Centrality (Structural)	
Degree	Betweeness
Delta	Egonet
Centrality (PageRank family)	
PageRank	Eigenvector
Gell point	Harmonic point
Centrality (Resolvent family)	
Resolvent series	Mercator series
Katz series	Neuman series
Community detection	
Newman-Girvan	Blondel
Walktrap	Fast Greedy
Shortest paths	
Dijkstra	Bellman-Ford
A^* algorithm	A^+ algorithm
Spanning tree	
Prim	Kruskal

Common metrics for ranking vertex centrality include PageRank, the eigenvector centrality [7], Gell point centrality [18], as well as resolvent centrality [11]. There have been also developed variants of these algorithms for fuzzy graphs [8]. Vertex set sizes can be estimated with techniques similar to those in [9]

NetworkX is an open access Python module for handling and storing in memory graphs and multilayer graphs. NumPy and SciPy are also Python modules implementing advanced mathematical functions for evaluating graph quantities among others.

F. Frailty Ontologies

Fraitly ontologies have been developed in order to describe many of the associated actors, events, and entities. Entities related to frailty include brain activity, psychological indicators, as well as basic physical status indicators such as blood pressure. Actors include the elderly, their familty, and medical experts and healthcare personel in general. Rules to validate the relationship between the entities or to discover new entities have been proposed.

G. MEG and EEG waveofmrs

MNE is an open access Python tool for processing EEG and MEG readings [12]. These readings can be used to represent and analyze brain activity, which is an important factor in assessing frailty. As both modes collect brain data over a time interval, their readings can be stored in a graph representing a brain map. Moreover, they can be used to build a second graph which represents events of interest. These events can be correlated with machine learning methods such as sparse neural networks [10] to events stored in frailty ontologies, facilitating brain activity abnormalities.

V. FUTURE WORK

This work can be expanded in a number of ways. A current trend in nearly every retrieval and data mining problem is that of multimodality, namely the combination of heterogeneous information in order to maximize the effectiveness of retrieval procedures. In the particular setting of medical retrieval multimodality translates to boosting text based retrieval with knowledge derived from other sources of medical data such as neuroimaging, EEG, or health statistics. In fact, since both neuroimaging and EEG data can be represented as graphs, they can be expected to be easily incorporated to Perseus.

Another research direction is medical multilingual information retrieval. There are many non-English public medical databases from where significant knowledge can be drawn, facilitating thus cooperation between researchers and allowing better coordination between field teams and practitioners. Moreover, multilingual retrieval allows quicker international response to severe disease outbreaks as incidents as well as potentially related symptoms recorded in non-English medical databases can be retrieved on behalf of an English speaking practitioner without prior translation. Additionally, advanced text semantic analysis can locate specific points of interest or data for such a user and even selectively translate part of the retrieved documents. As a caveat though, term polysemy can be aggravated within a multilingual context, as the same term can have different meanings in different languages.

ACKNOWLEDGEMENTS

This work was supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 690140.

REFERENCES

[1] A. R. Aronson. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association, 2001.

[2] M. Bada, D. Turi, R. McEntire, and R. Stevens. Using reasoning to guide annotation with gene ontology terms in GOAT. *ACM SIGMOD*, 33(2):27–32, 2004.

[3] R. Baud, A.-M. Rassinoux, and J. R. Scherrer. Natural language processing and semantical representation of medical texts. *Methods of Information in Medicine*, 31(2):117–125, 1992.

[4] D. Botstein, J. Cherry, M. Ashburner, C. Ball, J. Blake, H. Butler, A. Davis, K. Dolinski, S. Dwight, J. Eppig, et al. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

[5] G. O. Consortium et al. The gene ontology (go) database and informatics resource. *Nucleic acids research*, 32(suppl 1):D258–D261, 2004.

[6] G. Drakopoulos and A. Baroutiadi. Implementing centrality measures for Neo4j. In *Proceedings of the 8th ECESCON*, ECESCON 2015. ECESCON, April 2015.

[7] G. Drakopoulos, A. Baroutiadi, and V. Megalooikonomou. Higher order graph centrality metrics for Neo4j. In *Proceedings of the 6th International Conference on Information, Intelligence, Systemems, and Applications*, IISA 2015. IEEE, July 2015.

[8] G. Drakopoulos, A. Kanavos, C. Makris, and V. Megalooikonomou. On converting community detection algorithms for fuzzy graphs in Neo4j. In *Proceedings of the 5th International Workshop on Combinations of Intelligent Methods and Applications*, CIMA 2015. IEEE, November 2015.

[9] G. Drakopoulos, S. Kontopoulos, and C. Makris. Eventually consistent cardinality estimation with applications in biodata mining. In *Proceedings of the 31st Symposium on Applied Computing*, SAC 2016. ACM, April 2016.

[10] G. Drakopoulos and V. Megalooikonomou. On the weight sparsity of multilayer perceptrons. In *Proceedings of the 6th International Conference of Information, Intelligence, Systems, and Applications*, IISA 2015. IEEE, July 2015.

[11] E. Estrada and D. J. Higham. Network properties revealed through matrix functions. *SIAM Rev.*, 52(4):696–714, Nov. 2010.

[12] A. e. a. Gramfot. Meg and eeg data analysis with mne-python. *Frontiers in neuroscience*, 7(267), December 2013.

[13] S. Kontopoulos and G. Drakopoulos. A space efficient scheme for graph representation. In *Proceedings of the 26th International Conference on Tools with Artificial Intelligence*, ICTAI 2014, pages 299–303. IEEE, November 2014.

[14] S. Myhre, H. Tveit, T. Mollestad, and A. Lægreid. Additional gene ontology structure for improved biological reasoning. *Bioinformatics*, 22(16):2020–2027, 2006.

[15] I. Robinson, J. Webber, and E. Eifrem. *Graph Databases*. O'Reilly, June 2013.

[16] L. M. Schriml, C. Arze, S. Nadendla, Y.-W. W. Chang, M. Mazaitis, V. Felix, G. Feng, and W. A. Kibbe. Disease Ontology (DO): a backbone for disease semantic integration. *Nucleic acids research*, 40(D1):D940–D946, 2012.

[17] H. D. Tagare, C. C. Jaffe, and J. Duncan. Medical image databases: A content-based retrieval approach. *Journal of the American Medical Informatics Association*, 4(3):184–198, May 1997.

[18] H. Tong, B. A. Prakash, T. Eliassi-Rad, M. Faloutsos, and C. Faloutsos. Gelling and melting large graphs by edge manipulation. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, CIKM12, pages 245–254, New York, NY, USA, 2012. ACM.

[19] C.-H. Wei, C.-T. Li, and R. Wilson. A content-based approach to medical image database retrieval. *Database Modeling for Industrial Data Management: Emerging Technologies and Applications*, pages 258–290, 2005.