A Reasoning And Hypothesis-Generation Framework Based On Scalable Graph Analytics

Enabling Discoveries In Medicine Using Cray Urika-XA And Urika-GD

Sreenivas R. Sukumar, Larry W. Roberts and Jeffrey A. Graves
Oak Ridge National Laboratory, 1 Bethel Valley Road, Oak Ridge, TN 37831, USA.
Email: {sukumarsr@ornl.gov, robertslw@ornl.gov, gravesja@ornl.gov }

***Abstract*— Finding actionable insights from data has always been difficult. As the scale and forms of data evolve and morph, the task of finding value becomes even more challenging. Addressing, this challenge, data scientists at Oak Ridge National Laboratory are leveraging unique leadership infrastructure (e.g. Urika-XA and Urika-GD appliances) to develop scalable algorithms for semantic, logical and statistical reasoning with unstructured Big Data. In this paper, we present the deployment of such a framework called ORIGAMI (Oak Ridge Graph Analytics for Medical Innovations) on the National Library of Medicine's Semantic Medline (archive of medical knowledge since 1994). Medline contains over 70 million knowledge nuggets published in 23.5 million papers in medical literature with thousands more added each year. ORIGAMI is available as an open-science medical hypothesis generation tool - both as a web-service and an application programming interface (API) at http://hypothesis.ornl.gov .**

In 2015, ORIGAMI was featured in the Historical Clinical Pathological Conference in Baltimore as a demonstration of artificial intelligence to medicine and recognized as a Centennial Showcase Exhibit at the Radiological Society of North America (RSNA) Conference in Chicago. This paper describes the workflow built using the Cray Urika-XA and Urika-GD appliances that enables reasoning with the knowledge of every published medical paper every time a clinical researcher uses the ORIGAMI tool. Since becoming an online service, ORIGAMI has enabled clinical subject-matter experts to: (i) hypothesize the relationship between beta-blocker treatment and diabetic retinopathy; (ii) discover that xylene is an environmental cancer-causing carcinogen and (iii) aid doctors with diagnosis of challenging cases when rare diseases manifest with common symptoms.

Keywords-component; natural language reasoning; scalable graph analytics; hypothesis generation; semantic reasoning

I. Introduction

In 2014, researchers at the Oak Ridge National Lab (ORNL) and the National Library of Medicine (NLM) began efforts to build scalable informatics solutions to medical professionals aiming to solve clinically challenging cases. Computational linguists and information specialists that produce the Semantic Medline dataset at NLM were facing the following data science challenges – (i) the need to store, retrieve, parse and reason with massive knowledge graphs, (ii) the need to deal with datasets where there is more noise than signal, (iii) the computational complexity of dealing with hierarchies and semantic relationships while interpreting natural language, (iv) designing and applying algorithms that scale on modern compute architectures. We addressed the aforementioned challenges by leveraging ORNL's Compute and Data Environment of Science (CADES) that hosted the Urika-XA and Urika-GD infrastructure and previous effort from the authors (https://github.com/ssrangan/gm-sparql) that ported graph-theoretic algorithms to the Cray analytic architectures. The Urika-XA served as the extract, transform and data processing platform and Urika-GD acted as the interactive exploratory pattern search engine. Jointly, the workflow that involved data preparation and integration on the Urika-XA platform and the graph-analytic algorithms that scale on the shared memory Urika-GD platform enabled the design of a literature-based reasoning and hypothesis generation tool called ORIGAMI – Oak Ridge Graph Analytics for Medical Innovations.

Today, ORIGAMI helps "connect the dots" across predications provided in Medline, allows a medical expert to explore non-obvious clinical associations with semantic meaning and generate a significance score of belief for a "hypothesized" association. ORIGAMI is able to conduct searches based on numerous information foraging heuristics on 70 million predications in the order of a few seconds rather than the years it would take to manually perform this search over the entire body of knowledge. The rest of the paper describes the background behind this effort and the building blocks of the scalable knowledge-reasoning and hypothesis-generation framework.

II. Background

A. *Artificial Intelligence in Medicine*

The fascination of applying AI to medicine dates back to the late 1950s [1] followed by several academic publications documenting collaborations of doctors and computer scientists [2]. The philosophical foundation with both disciplines being the process of collecting data and applying inference rules to make a predictive diagnosis of a disease. Ledley and Lusted [1] pointed out that medical reasoning was not magic but instead contained well-recognized inference strategies: Boolean logic, symbolic inference, and Bayesian probability. Several tools such as PROMIS [3], CASNET [4], MYCIN [5], QMR [6], INTERNIST [7], DXPLAIN [8] and ILIAD [9] have become available since then. The performance of these programs are evaluated, validated and compared by running them on some challenging case reports (called clinicopathological cases, or CPCs) such as those that appear each week in the New England Journal of Medicine. The performance analyses of these tools routinely outperformed medical students in training to be physicians. However, due to the lack of digital interoperability and standards in representing health records and the exploding nature of medical research, the computer-based expert systems were unable to sustain the momentum. Also, expert systems that demonstrated the ability to get better at diagnosing typical/common cases were unable to handle rare or mysterious illnesses.

The effort to normalize and archive medical knowledge with interoperable standard terminologies led to the Unified Medical Language System (UMLS), a project at the National Library of Medicine with the goal of integrating a number of existing medical vocabularies using a common semantic structure [10] and Semantic Medline, a semantic database of biomedical research [11]. ORIGAMI was inspired by seminal work by D.R. Swanson [12] that leverages UMLS and Semantic Medline. Swanson's work began a new discipline of AI and its application to medicine called "literature-based discovery". Literature-based discovery is the process by which academic publications are linked creatively to find new relationships across existing knowledge [13]. Unlike discovery in the empirical sciences, where laboratory experiments create new knowledge, Literature-based discovery seeks to connect existing knowledge from empirical results by bringing to light relationships that are implicated and neglected [14-15]. ARROWSMITH – a tool developed by Swanson after discovering connections between Migraine and Magnesium; Fish-oil and Raynaud's disease; Somatomedin C and arginine and many such associations was the first software prototype that demonstrated the convergence of digital search and the utility of linking diverse knowledge areas for literature-based discovery. While the methods in ARROWSMITH are robust, the ability to scale to the increasing body of knowledge was a challenge.

The MEDLINE dataset in 2014 consists of 70 million predications from 23.5 million publications. The number of neglected connections between Migraine and Magnesium in the 2014 knowledgebase is approximately 133,000. In other words, if a researcher wanted to emulate Swanson, he or she would have to sift through the 133,000 possibilities to discover the 11 most relevant ones Swanson found in 1987. This becomes an increasingly harder problem for humans and thankfully a manageable one due to recent developments in scalable graph analytics. ORIGAMI was motivated toward addressing the following questions – (i) Can we build tools and infrastructure that enable knowledge discovery from large noisy datasets? (ii) Can we implement scalable algorithms that can intelligently learn to reason with text and successfully deal with data even when noise overwhelms the signal? (iii) Can such a system be flexible and adaptive to the evolving body of medical knowledge?

B. Convergence of scalable graph analytics, artificial intelligence and medicine

ORIGAMI is the solution that answers the aforementioned questions through the convergence of scalable graph analytics and artificial intelligent heuristics. This convergence enables scalable algorithms for semantic, logical and statistical reasoning with Big Data (i.e., data stored in databases as well as unstructured data in documents). The functionality of ORIGAMI is comparable to the IBM's Watson technology that won Jeopardy! on television [16]. IBM's Watson digested large volumes of unstructured text and then retrieved contextually meaningful results using machine learning techniques. The IBM Watson approach to cognitive computing is built on strong natural language processing of unstructured documents and the organization and staging of the content into high performance computing platforms that are able to speed up rule-based inferencing and retrieval. ORIGAMI on the other hand is a reasoning and hypothesis generation framework founded on information foraging principles that leverages the graph data structure to learn and discover patterns. Graph structures offer an intuitive representation for link analysis, pattern search and discovery. ORIGAMI leverages the ability to do both graph pattern-matching and graph-theoretic mining on datasets at similar latencies [17] (in the order of a few seconds) and the marriage of graph-theoretic mining and the semantic web technologies [18] at scale on massive heterogeneous graph structures [19-21]. It is founded on the idea that knowledge constantly evolves and that the reasoning framework should be flexible in accommodating the evolution. ORIGAMI accepts data in most common forms (SQL, Text, etc.) transforms and ingests them as W3C standard machine-readable RDF triples. It uses RDF-triple stores as the backend host for the data that can be queried using the SPARQL query language. By hosting the RDF store on the Urika-GD platform that is a Threadstorm based shared-memory architecture supercomputer and implementing algorithms that scale on this architecture we augment the ad-hoc query ability using SPARQL with mathematically founded graph-theoretic algorithms [21-24].

Our approach is different from IBM's approach in the following ways: (i) ORIGAMI because of its link prediction algorithms can hypothesize potential associations that are not obvious or explicit – i.e., it can make educated guesses about an association as opposed to retrieving a pre-recorded association. (ii) ORIGAMI learns on the fly by making mistakes on noisy data– i.e., The parallel nature of the query running on a shared-memory architecture evaluates several thousand answers during the same time it takes for other architectures to retrieve one result. (iii) ORIGAMI is a no-index exhaustive-searching divergent thinker – i.e., learns structure and saliency automatically from the data to produce a salient result-set without the bias of model-fitting with the machine learning approach of IBM's Watson. Every query to ORIGAMI touches the entire dataset hosted in memory before an answer is generated. Currently we are able to deal with up to 2 TBs on the Urika-GD platform. In the following sections, we provide the technical and mathematical details behind ORIGAMI.

III. APPROACH

A. *Data*

The Semantic Knowledge Representation (SKR) project at the National Library of Medicine, conducts basic research in symbolic natural language processing. The SKR project maintains a database of 70 million predications (subject-predict-object sentences) extracted from publications archived in the PubMed database. This database called SEMMED (short for Semantic Medline and available at http://skr3.nlm.nih.gov/SemMedDB/dbinfo.html) is the data source for the ORIGAMI application.

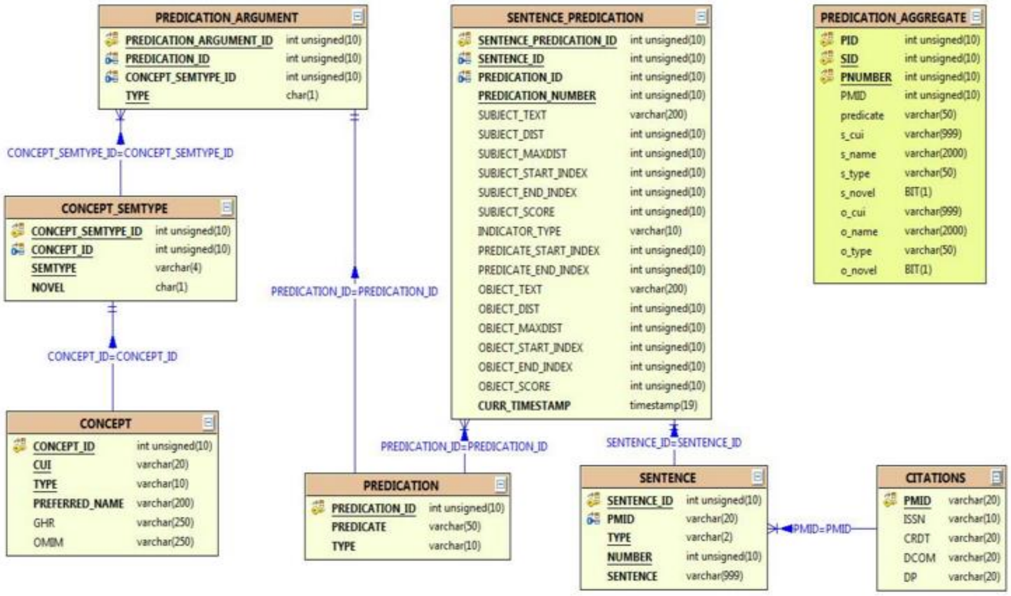
The database from National Library of Medicine is available for public download in SQL format. Of primary use was the PREDICATION_AGGREGATE table shown in Figure 1, which contains a summary of fields for efficient access. Those fields include predicate, subject name, subject type, object name and object type. We converted the SQL format to Resource Description Framework (RDF) format triples. An example triple is <influenza> <ISA> <Acute_viral_disease>. The result of the conversion is an "nq" file that contains all 70 million predications, which was then uploaded on Urika-GD. The extract, transform and load tasks are typically done using the software mentioned in [26]. The Urika-XA machine was used for he data transformation. Figure 2 illustrates the data preparation process.

B. *Algorithms*

In this section we describe the algorithms that reason with the data once loaded into the Urika-GD platform. We begin with the formal mathematical definitions before explaining the algorithms.

Definitions

Let L be a collection of labels, V be a nonempty set of vertices, $E = V \text{ x } V$ be a multiset of ordered pairs of vertices called edges, $\varphi_V : V \rightarrow L$ be a function assigning labels to vertices, and $\varphi_E : V \rightarrow L$ be a function assigning labels to edges. Then, $G = (V, E, \varphi_V, \varphi_E)$ is a directed labeled multigraph with loop edges. The data is not directly represented as a directed labeled multigraph, but rather as a set of triples of the form subject-predicate-object in the Resource Description Framework (RDF). A nonempty collection of triples leads to a natural representation of a directed labeled multigraph with possible loop edges, and it is often useful to change between these two views. Each distinct subject/object in the collection is represented by a single vertex, and the label of a vertex is the value of the subject/object it represents. This implies that v is a surjective (one-to-one) function. Furthermore, since the data is represented as a *set* of triples, it is possible to have multiple directed edges with the same label, but no pair of



PREDICATION_ARGUMENT	
PREDICATION_ARGUMENT_ID	int unsigned(10)
PREDICATION_ID	int unsigned(10)
CONCEPT_SEMTYPE_ID	int unsigned(10)
TYPE	char(1)

CONCEPT_SEMTYPE	
CONCEPT_SEMTYPE_ID	int unsigned(10)
CONCEPT_ID	int unsigned(10)
SEMTYPE	varchar(4)
NOVEL	char(1)

CONCEPT	
CONCEPT_ID	int unsigned(10)
CUI	varchar(20)
TYPE	varchar(10)
PREFERRED_NAME	varchar(200)
GHR	varchar(250)
OMIM	varchar(250)

SENTENCE_PREDICATION	
SENTENCE_PREDICATION_ID	int unsigned(10)
SENTENCE_ID	int unsigned(10)
PREDICATION_ID	int unsigned(10)
PREDICATION_NUMBER	int unsigned(10)
SUBJECT_TEXT	varchar(200)
SUBJECT_DIST	int unsigned(10)
SUBJECT_MAXDIST	int unsigned(10)
SUBJECT_START_INDEX	int unsigned(10)
SUBJECT_END_INDEX	int unsigned(10)
SUBJECT_SCORE	int unsigned(10)
INDICATOR_TYPE	varchar(10)
PREDICATE_START_INDEX	int unsigned(10)
PREDICATE_END_INDEX	int unsigned(10)
OBJECT_TEXT	varchar(200)
OBJECT_DIST	int unsigned(10)
OBJECT_MAXDIST	int unsigned(10)
OBJECT_START_INDEX	int unsigned(10)
OBJECT_END_INDEX	int unsigned(10)
OBJECT_SCORE	int unsigned(10)
CURR_TIMESTAMP	timestamp(19)

PREDICATION	
PREDICATION_ID	int unsigned(10)
PREDICATE	varchar(50)
TYPE	varchar(10)

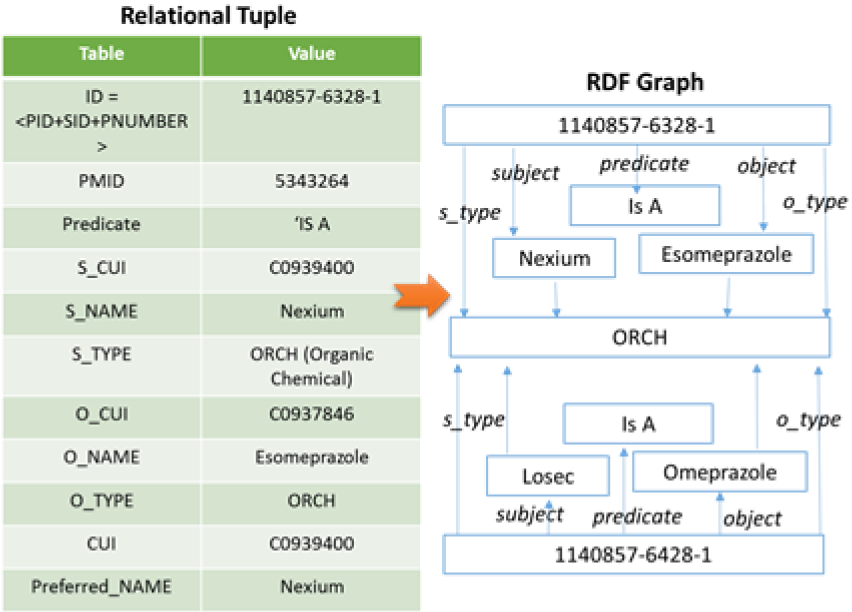
SENTENCE	
SENTENCE_ID	int unsigned(10)
PMID	varchar(20)
TYPE	varchar(2)
NUMBER	int unsigned(10)
SENTENCE	varchar(999)

PREDICATION_AGGREGATE	
PID	int unsigned(10)
SID	int unsigned(10)
PNUMBER	int unsigned(10)
PMID	int unsigned(10)
predicate	varchar(50)
s_cui	varchar(999)
s_name	varchar(2000)
s_type	varchar(50)
s_novel	BIT(1)
o_cui	varchar(999)
o_name	varchar(2000)
o_type	varchar(50)
o_novel	BIT(1)

CITATIONS	
PMID	varchar(20)
ISSN	varchar(10)
CRDT	varchar(20)
DCOM	varchar(20)
DP	varchar(20)

CONCEPT_SEMTYPE_ID=CONCEPT_SEMTYPE_ID
PREDICATION_ID=PREDICATION_ID
CONCEPT_ID=CONCEPT_ID
PREDICATION_ID=PREDICATION_ID
SENTENCE_ID=SENTENCE_ID
PMID=PMID

Figure 1. The entity relationship diagram of the Semantic Medline (SEMMED) dataset from the National Library of Medicine. In addition to the database being a list of medical predications, each knowledge nugget from the PubMed archive is associated with the provenance of which paper contributed to that predication.



Relational Tuple

Table	Value
ID = <PID+SID+PNUMBER >	1140857-6328-1
PMID	5343264
Predicate	'IS A
S_CUI	C0939400
S_NAME	Nexium
S_TYPE	ORCH (Organic Chemical)
O_CUI	C0937846
O_NAME	Esomeprazole
O_TYPE	ORCH
CUI	C0939400
Preferred_NAME	Nexium

Figure 2. The relational tuple from the multi-tabular join in Figure 1 is converted into a W3C standard structure as a triple. In this example, the two predications are 'Nexium is a Esomerprazole' and 'Losec is a Omeprazole'. The subject and object terms are associated with meta-data term 'ORCH" short for organic chemical. Each of the 70 million predications in Semantic Medlne dataset is converted into this RDF Graph structure of subjects, predicates, objects and atrributes. The data conversion software is implemented using the Map-Reduce model that scales well on the Urika-XA.

vertices can have multiple directed edges between them with the same label; multiple edges between a pair vertices is permitted so long as all of the edge labels are distinct. Since the data is a collection of triples and a triple defines an edge, a vertex in V must be adjacent to at least one other vertex (i.e., $\forall_v \in V$, $\exists$ (a,b) $\in$ E such that $v = a$ or $v = b$). From this point forward, unless stated otherwise, the word graph will be used to refer to a directed labeled multigraph with possible loop edges that represent a set of triples, and the two representations will be used interchangeably.

A walk in a graph $G = (V, E, \varphi_V, \varphi_E)$ is a sequence of alternating vertices and edges that begins and ends with a vertex such that for each edge, the source is the preceding vertex and the destination is the following vertex. The length of a walk is defined to be the number of edges in the walk. A single vertex-edge-vertex contiguous subsequence of a walk is often referred to as a hop and a walk of length n is sometimes referred to as an n-hop walk. A walk in which every edge is unique is called a trail. If the first and last vertexes in a trail are the same, the trail is called a circuit. A walk in which every vertex is unique is called a path; a walk in which every vertex in unique except the first vertex and the last vertex is called a cycle.

Saliency Estimation Algorithm

The saliency estimation algorithms extract graph-theoretic metrics to understand the RDF graph and associate a probabilistic score of saliency for each triple in the graph. This is done using the statistics of the conditional distributions around subject and predicate terms and their meta-data attribute types. For each triple in the SemMedDB graph, a score and a threshold value is calculated. If the score of the triple does not exceed the threshold of the triple, the triple is considered salient and labelled accordingly. The score for a triple, (subject, predicate, object), is defined to be the number of times the subject-predicate appears in the graph multiplied by the number of times the predicate-object appears in the graph. More formally, if

$$score_{sp}(sub,pred) = |\{(sub,pred,o) : (sub,pred,o) \in G\}|$$

and

$$score_{po}\ (pred,obj) = |\{(s,pred,obj) : (s,pred,obj) \in G\}|$$

then,

$$score(sub,pred,obj) = score_{sp}(sub,pred) \bullet score_{po}(pred,obj).$$

The threshold for the triple is defined to be the average of the subject-predicate pair counts times the average of the predicate-object pair counts, where the predicate is given in the triple and the subject and object range over all possible subjects and objects in the graph. More specifically, if

$$P_S = \{(\text{sub,pred}) : (\text{sub,pred},o) \in \text{G}\},\ P_O = \{(\text{pred,obj}) : (s,\text{pred,obj}) \in G\},$$

and if

$$\text{threshold}_{sp}(\text{pred}) = \frac{\sum_{(\text{sub,pred}) \in P_S} \text{score}(\text{sub,pred})}{|P_S|}$$

and

$$\text{threshold}_{po}(\text{pred}) = \frac{\sum_{(\text{pred,obj}) \in P_O} \text{score}(\text{pred,obj})}{|P_O|}$$

then the threshold is given by

$$\text{threshold(pred)} = \text{threshold}_{sp}\text{(pred)} \bullet (\text{threshold}_{po}\text{(pred)}.$$

The salient graph *G’* is taken to be the collection of all triples in the graph *G* such that the score of the triple is greater than or equal to the threshold of the triple. That is to say,

$$G' = \{(sub,pred,obj) \in G : score(sub,pred,obj) \geq threshold(sub,pred,obj)\}$$

The saliency estimation algorithms allow us to cache the important triples in memory for further reasoning and avoid having to read from disk during query execution.

Term Reasoning Algorithms

The objective of the term-reasoning algorithms is the following: Given a specific search/query term, one may wish to uncover a collection of similar or contextually relevant terms. We define a few heuristics of term similarity based on the predicate relevance, meta-pattern structure and predication-pattern similarity.

Predicate Relevance: Given a desired term, the predicate relevance heuristic can be used to retrieve and explore terms that are closely related to the given term. This heuristic allows the user to navigate the SemMedDB graph by choosing predicate relevance. Specifically, a query using this heuristic returns a collection of *n* hop paths from the query term. A score is calculated for each path by taking the reciprocal of the score of each subject-predicate-object triple represented in the path. For example, let $p = (v_0,e_0,v_1,e_1,\ldots,e_{n-1},v_n)$ be an arbitrary n hop path. Then, the score of path p is given by

$$\frac{\sum_{i=0}^{n-1} \text{score}(v_i, e_i, v_{i+1})^{-1}}{n}$$

Note that each (v_i,e_i,v_{i+1}) for $0 \leq i \leq n$ is a triple in the graph. These scores are then used to order the paths from highest to lowest score.

Specific Relevance: The specific reasoning application takes the view that two terms are similar if they have a similar neighborhood in the SemMedDB knowledge graph. Given a term of interest, the size of the overlap (intersection) between the neighbors of the term of interest and the neighborhoods of every other term in the graph is calculated. The top *n* terms with the largest overlap are returned to the user as the most similar terms. More specifically, let $G = (V, E, \varphi_V, \varphi_E)$ be a graph and let $N_G^+(v) = \{u: (v, u) \in E\}$, let $N_G^-(v) = \{u: (v, u) \in E\}$, and $N_G(v) = N_G^+(v) \cup N_G^-(v)$ denote the (open) out-neighborhood, (open) in-neighborhood, and (open) neighborhood of a vertex *v* in *V*, respectively. Then the similarity between two vertices *v* and *u* is defined to be $|N_G(v) \cap N_G(u)|$. Note that the labels of the edges connecting a term to its neighbors are ignored when considering the neighborhood.

Pattern Relevance: Much like specific reasoning, the pattern similarity heuristic views two terms as similar if they have similar out-neighborhoods in the SemMedDB graph. Unlike specific reasoning, the value of the predicate is viewed as important when measuring similarity. If we view our graph G as a collection of subject-predicate-object triples, then the similarity between two vertices *v* and *u* is defined to be $|\{(p,s) : (v,p,s) \in G\} \cap \{(p,s) : (u,p,s) \in G\}|$. This heuristic helps find graph sub-structures that share the same predicate-object associations as the query term. The search is done over the entire knowledge graph in memory.

Path Reasoning Algorithms

Given a pair of search terms, one may wish to evaluate the strength of the association between the two terms or explore the context of terms that relates the two search terms. The mathematical formulae for retrieving context terms and paths-of-interest between two search terms are described below.

Context Relevance: Suppose *u* and *v* are vertices in a graph *G* and

$$V = (\bigcup_{v' \in N_G^+(v)} N_G^+(v')) \cup N_G^+(v)$$

and

$$U = (\bigcup_{u' \in N_G^+(u)} N_G^+(u')) \cup N_G^+(u)$$

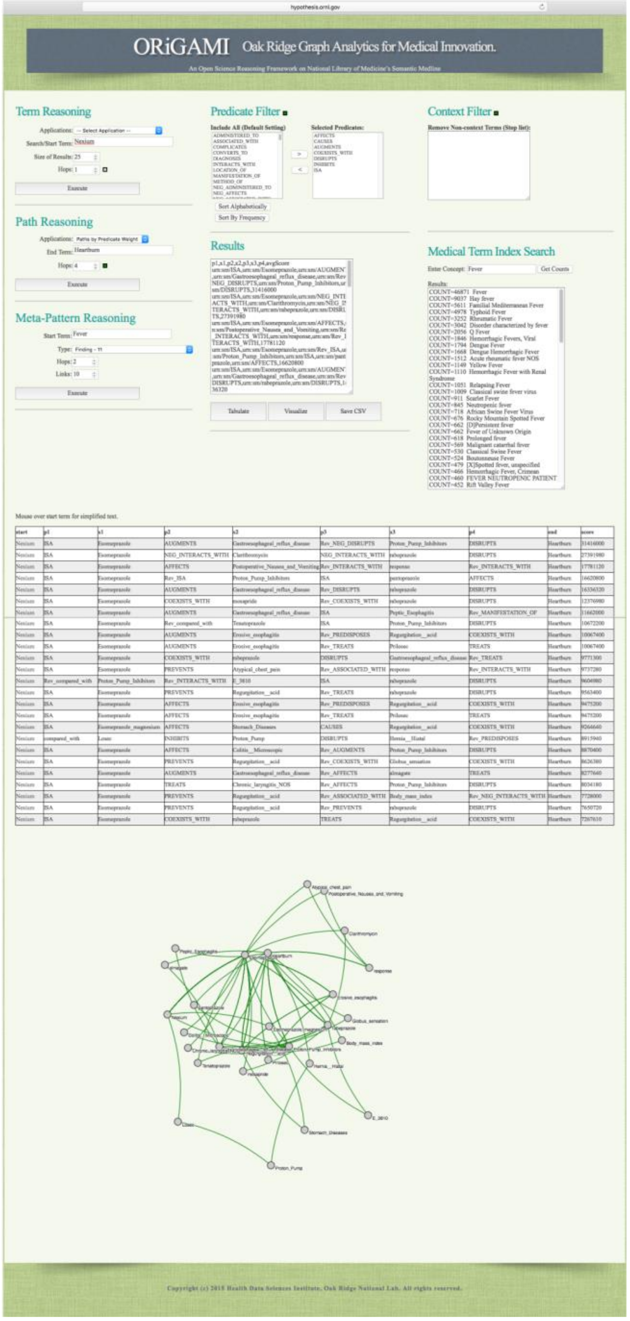
are the collection of *k*-hop neighbors of *v* and *u*, respectively, then the context relevant terms are defined by the vertices in $V \cap U$. For the sake of simplicity, the context terms are the overlap between the set of all one and two hops neighbors of the specified terms in the SemMedDB graph in ORIGAMI although we are able to do up to a 5-hop radius in a few seconds.

Path Extraction: The path extraction algorithms retrieve a list of shortest paths within *k*-hops ($k < 8$) between a specified start and end term in the SemMedDB graph. Using the predicate relevance score that ranked each triple in the knowledge graph, we are able to compute a path saliency score as the sum or product of the individual triple saliency scores in the path. Logical relevance is imposed during the path extraction by associating each predicate with a weight. These weights can be learned automatically or arbitrarily specified by a subject matter expert. The weights are used to map the logical relevance of the co-occurrence of 1 or more predicates. The saliency score of each retrieved path is the product of the weights of the predicates in the path.

Meta-Pattern Relevance: This heuristic extracts paths and scores them similar to the path extraction algorithm described above. The difference is that the paths are extracted about the meta-data attributes of the subjects, predicates and objects. This is done using a random-walk formulation for exploration of meta-structure about terms of interest and the walks are biased based on the conditional probabilistic models extracted using the saliency estimation heuristics. This allows for ORIGAMI to intelligently understand the structure and statistics of the meta-data before actually having to evaluate every path.

C. *User-Interface*

A screenshot of ORIGAMI's web-interface is shown below in Figure 3. The saliency extraction, term and path reasoning algorithms are all available as executables and APIs. The interface allows interactive querying, visualization and exploratory investigation of associations.



Term Reasoning

Applications: -- Select Application --

Search/Start Term: Nexium

Size of Results: 25

Hops: 1

Path Reasoning

Applications: Paths by Predicate Weight

End Term: Heartburn

Hops: 4

Meta-Pattern Reasoning

Start Term: Fever

Type: Finding - T1

Hops: 2

Links: 10

Predicate Filter

Include All (Default Setting):

```
ADMINISTERED_TO
ASSOCIATED_WITH
COMPLICATES
CONVERTS_TO
DIAGNOSES
INTERACTS_WITH
LOCATION_OF
MANIFESTATION_OF
METHOD_OF
NEG_ADMINISTERED_TO
NEG_AFFECTS
```

Selected Predicates:

```
AFFECTS
CAUSES
AUGMENTS
COEXISTS_WITH
DISRUPTS
INHIBITS
ISA
```

Results

```
p1,x1,p2,x2,p3,x3,p4,avgScore
urn:sm/ISA,urn:sm/Esomeprazole,urn:sm/AUGMEN
,urn:sm/Gastroesophageal_reflux_disease,urn:sm/Rev
NEG_DISRUPTS,urn:sm/Proton_Pump_Inhibitors,ur
sm/DISRUPTS,31416000
urn:sm/ISA,urn:sm/Esomeprazole,urn:sm/NEG_INTE
ACTS_WITH,urn:sm/Clarithromycin,urn:sm/NEG_IN
TERACTS_WITH,urn:sm/rabeprazole,urn:sm/DISRU
TS,27391980
urn:sm/ISA,urn:sm/Esomeprazole,urn:sm/AFFECTS,
n:sm/Postoperative_Nausea_and_Vomiting,urn:sm/Re
_INTERACTS_WITH,urn:sm/response,urn:sm/Rev_I
TERACTS_WITH,17781120
urn:sm/ISA,urn:sm/Esomeprazole,urn:sm/Rev_ISA,u
sm/Proton_Pump_Inhibitors,urn:sm/ISA,urn:sm/pant
prazole,urn:sm/AFFECTS,16620800
urn:sm/ISA,urn:sm/Esomeprazole,urn:sm/AUGMEN
,urn:sm/Gastroesophageal_reflux_disease,urn:sm/Rev
DISRUPTS,urn:sm/rabeprazole,urn:sm/DISRUPTS,1
36320
```

Context Filter

Remove Non-context Terms (Stop list):

Medical Term Index Search

Enter Concept: Fever

Results:

```
COUNT=46871  Fever
COUNT=9037  Hay fever
COUNT=5611  Familial Mediterranean Fever
COUNT=4978  Typhoid Fever
COUNT=3252  Rheumatic Fever
COUNT=3042  Disorder characterized by fever
COUNT=2056  Q Fever
COUNT=1846  Hemorrhagic Fevers, Viral
COUNT=1794  Dengue Fever
COUNT=1668  Dengue Hemorrhagic Fever
COUNT=1512  Acute rheumatic fever NOS
COUNT=1149  Yellow Fever
COUNT=1110  Hemorrhagic Fever with Renal
Syndrome
COUNT=1051  Relapsing Fever
COUNT=1009  Classical swine fever virus
COUNT=911  Scarlet Fever
COUNT=845  Neutropenic fever
COUNT=718  African Swine Fever Virus
COUNT=676  Rocky Mountain Spotted Fever
COUNT=662  [D]Persistent fever
COUNT=662  Fever of Unknown Origin
COUNT=618  Prolonged fever
COUNT=569  Malignant catarrhal fever
COUNT=530  Classical Swine Fever
COUNT=524  Boutonneuse Fever
COUNT=479  [X]Spotted fever, unspecified
COUNT=466  Hemorrhagic Fever, Crimean
COUNT=460  FEVER NEUTROPENIC PATIENT
COUNT=452  Rift Valley Fever
```

Mouse over start term for simplified text.

start	p1	x1	p2	x2	p3	x3	p4	end	score
Nexium	ISA	Esomeprazole	AUGMENTS	Gastroesophageal_reflux_disease	Rev_NEG_DISRUPTS	Proton_Pump_Inhibitors	DISRUPTS	Heartburn	31416000
Nexium	ISA	Esomeprazole	NEG_INTERACTS_WITH	Clarithromycin	NEG_INTERACTS_WITH	rabeprazole	DISRUPTS	Heartburn	27391980
Nexium	ISA	Esomeprazole	AFFECTS	Postoperative_Nausea_and_Vomiting	Rev_INTERACTS_WITH	response	Rev_INTERACTS_WITH	Heartburn	17781120
Nexium	ISA	Esomeprazole	Rev_ISA	Proton_Pump_Inhibitors	ISA	pantoprazole	AFFECTS	Heartburn	16620800
Nexium	ISA	Esomeprazole	AUGMENTS	Gastroesophageal_reflux_disease	Rev_DISRUPTS	rabeprazole	DISRUPTS	Heartburn	16336320
Nexium	ISA	Esomeprazole	COEXISTS_WITH	mosapride	Rev_COEXISTS_WITH	rabeprazole	DISRUPTS	Heartburn	12376980
Nexium	ISA	Esomeprazole	AUGMENTS	Gastroesophageal_reflux_disease	ISA	Peptic_Esophagitis	Rev_MANIFESTATION_OF	Heartburn	11662000
Nexium	ISA	Esomeprazole	Rev_compared_with	Tenatoprazole	ISA	Proton_Pump_Inhibitors	DISRUPTS	Heartburn	10672200
Nexium	ISA	Esomeprazole	AUGMENTS	Erosive_esophagitis	Rev_PREDISPOSES	Regurgitation__acid	COEXISTS_WITH	Heartburn	10067400
Nexium	ISA	Esomeprazole	AUGMENTS	Erosive_esophagitis	Rev_TREATS	Prilosec	TREATS	Heartburn	10067400
Nexium	ISA	Esomeprazole	COEXISTS_WITH	rabeprazole	DISRUPTS	Gastroesophageal_reflux_disease	Rev_TREATS	Heartburn	9771300
Nexium	ISA	Esomeprazole	PREVENTS	Atypical_chest_pain	Rev_ASSOCIATED_WITH	response	Rev_INTERACTS_WITH	Heartburn	9737280
Nexium	Rev_compared_with	Proton_Pump_Inhibitors	Rev_INTERACTS_WITH	E_3810	ISA	rabeprazole	DISRUPTS	Heartburn	9604980
Nexium	ISA	Esomeprazole	PREVENTS	Regurgitation__acid	Rev_TREATS	rabeprazole	DISRUPTS	Heartburn	9563400
Nexium	ISA	Esomeprazole	AFFECTS	Erosive_esophagitis	Rev_PREDISPOSES	Regurgitation__acid	COEXISTS_WITH	Heartburn	9475200
Nexium	ISA	Esomeprazole	AFFECTS	Erosive_esophagitis	Rev_TREATS	Prilosec	TREATS	Heartburn	9475200
Nexium	ISA	Esomeprazole_magnesium	AFFECTS	Stomach_Diseases	CAUSES	Regurgitation__acid	COEXISTS_WITH	Heartburn	9266640
Nexium	compared_with	Losec	INHIBITS	Proton_Pump	DISRUPTS	Hernia__Hiatal	Rev_PREDISPOSES	Heartburn	8915940
Nexium	ISA	Esomeprazole	AFFECTS	Colitis__Microscopic	Rev_AUGMENTS	Proton_Pump_Inhibitors	DISRUPTS	Heartburn	8870400
Nexium	ISA	Esomeprazole	PREVENTS	Regurgitation__acid	Rev_COEXISTS_WITH	Globus_sensation	COEXISTS_WITH	Heartburn	8626380
Nexium	ISA	Esomeprazole	AUGMENTS	Gastroesophageal_reflux_disease	Rev_AFFECTS	alinagate	TREATS	Heartburn	8277640
Nexium	ISA	Esomeprazole	TREATS	Chronic_laryngitis_NOS	Rev_AFFECTS	Proton_Pump_Inhibitors	DISRUPTS	Heartburn	8034180
Nexium	ISA	Esomeprazole	PREVENTS	Regurgitation__acid	Rev_ASSOCIATED_WITH	Body_mass_index	Rev_NEG_INTERACTS_WITH	Heartburn	7728000
Nexium	ISA	Esomeprazole	PREVENTS	Regurgitation__acid	Rev_PREVENTS	rabeprazole	DISRUPTS	Heartburn	7650720
Nexium	ISA	Esomeprazole	COEXISTS_WITH	rabeprazole	TREATS	Regurgitation__acid	COEXISTS_WITH	Heartburn	7267610

Copyright (c) 2015 Health Data Sciences Institute, Oak Ridge National Lab. All rights reserved.

Figure 3. The ORIGAMI web-interface allows users to explore medical literature. Experts and novices alike are able to search the medical index of terms from UMLS to begin their exploration and each of the algorithms presented in this paper can then be executed as an application on the entire Semantic Medline dataset.

IV. CASE STUDY

The utility of ORIGAMI was demonstrated during University of Maryland's 2015 Historical Clinicopathological Conference (http://medicalalumni.org/historicalcpc/home/) in collaboration with Dr. Elliot Siegel. This event, organized by Dr. Philip Mackowiak, is a medical clinical exercise (conducted every year for the last 22 years) in which a historical celebrity patient (circa 19th century) is presented to a panel of experienced clinicians for discussion. The mystery patient's case is presented by a historian based on his research from multiple biographies written about the celebrity. The panel of doctors then discuss what disease(s) did the celebrity suffer from and how this person died – based on symptomatic descriptions from the biographical summary. The doctors then argue how they would treat someone walking into their clinic with a similar medical history as the celebrity. This year, as proof of progress in artificial intelligence, the conference organizers requested a demonstration of how a tool like ORIGAMI can help a doctor diagnose a "rare disease" case by using our supercomputers to generate a ranked list of hypotheses for the cause of death of the celebrity "Oliver Cromwell – also known as the Terror of Europe".

The workflow developed to tackle this challenge was as follows. The initial step was to map out terminology presented in the 'Case' to the National Library of Medicine's PubMed Medical Subject Headlines (MeSH), e.g. "fever maps out to over 100 "standard" terms in NLM. Multiple heuristics were used to provide meaning and context based on relationship patterns around the term of interest; e.g. looking at the term "fever", diseases with similar patterns and similar meanings are added as context. Based on all the terms in the 'Case' description, a case context is generated. We note that case contexts can use even seemingly medically irrelevant patient information; e.g. for a patient with Welsh ancestry or being a soldier. Being Welsh would reveal the medical pre-disposition to a list of diseases, proteins, genes, etc. and being a soldier would create a context term such as post-traumatic disorder. The relationship of MeSH terms in the 23 million plus articles that are "read" by the supercomputer are then presented visually for the subject matter expert to interact and prioritize. The doctors can then conduct co-occurrence analysis and execute association rule-discovery algorithms that return results in the order of a few seconds. We then compute the probability of the symptoms being associated with a particular disease using the random walk algorithm. We task our shared-memory machine to conduct two batches of random walks (i) from random symptoms towards diseases (ii) from 'Case' symptoms towards diseases. We derived a scoring mechanism to evaluate paths when terms specific to the 'Case' are involved in the walk in either case. By initiating thousands of such random walks (which would be computationally impossible without the shared-memory architecture) in parallel – we create potential hypothesis based on weak-but very relevant associations across the 70 million predications. We filter for relevance by ignoring paths common to both the ‘Case’ walks and the random symptom-disease walks. The scoring mechanism helps us associate a probability to the disease hypothesis.

In the case of the historical patient Oliver Cromwell, the hypotheses generated by ORIGAMI pleasantly surprised the doctors. ORIGAMI found, with the highest probability, the same primary diagnosis (Malaria) as an expert opinion - instead of taking many weeks/months, the differential diagnosis was made in seconds. ORIGAMI provided other possible diagnoses in order of probability allowing the domain expert to drill down into the reasons that a particular diagnosis was made. Additional possibilities with lower probabilities included Staphylococcal Bacteremia, Urinary Tract Infection, Poisoning Syndrome and Coccidiosis. On a crowd-sourced independent survey of doctors not in the panel, Malaria was the consensus diagnosis with a few doctors arguing for many of the lower-probability diagnoses generated by ORIGAMI.

V. SUMMARY

ORIGAMI is an artificial intelligence system for the discovery and ranking of meaningful associations by reasoning with unstructured text documents (literature) critical to designing domain-specific hypotheses. As a software application it is a suite of scalable algorithms for semantic, logical and statistical reasoning with Big Data (i.e., data stored in databases as well as unstructured data in documents) that scale on Cray analytics hardware Urika-XA and Urika-GD. This technology is a futuristic next-generation knowledge-discovery framework that is: (a) knowledge nurturing (i.e., evolves seamlessly with newer knowledge and data), (b) smart and curious (i.e., using natural language processing, intelligent data parsing and harmonization, information-foraging and reasoning algorithms to digest content) and (c) synergistic enabling computer-assisted serendipity (i.e., interfaces computers with what they do best to help subject-matter-experts do their best.

ACKNOWLEDGMENT

This manuscript has been co-authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doepublic-access-plan).

The authors thank Thomas Rindflesch, Marcelo Fisczman and Mike Carielli from the National Library of Medicine for their invaluable inputs on Semantic Medline and Dr. Elliot Siegel from the University of Maryland for helping us with the medical case studies.

REFERENCES

[1] Ledley, R.S. and Lusted, L.B., 1959. Reasoning Foundations of Medical Diagnosis Symbolic logic, probability, and value theory aid our understanding of how physicians reason. Science, 130(3366), pp.9-21.

[2] Altman, R.B., 1999. AI in medicine: The spectrum of challenges from managed care to molecular medicine. AI magazine, 20(3), p.67.

[3] Tufo, H.M., Bouchard, R.E., Rubin, A.S., Twitchell, J.C., VanBuren, H.C., Weed, L.B. and Rothwell, M., 1977. Problem-oriented approach to practice: I. Economic impact. JAMA, 238(5), pp.414-417.

[4] Kulikowski, C.A. and Weiss, S.M., 1982. Representation of expert knowledge for consultation: the CASNET and EXPERT projects. Artificial Intelligence in medicine, 51.

[5] Buchanan, B.G. and Shortliffe, E.H. eds., 1984. Rule-based expert systems (Vol. 3). Reading, MA: Addison-Wesley.

[6] Miller, R.A.., Masarie, F.E. and Myers, J.D., 1985. Quick medical reference (QMR) for diagnostic assistance. MD computing: computers in medical practice, 3(5), pp.34-48.

[7] Miller, R.A., Pople Jr, H.E. and Myers, J.D., 1982. Internist-I, an experimental computer-based diagnostic consultant for general internal medicine. New England Journal of Medicine, 307(8), pp.468-476.

[8] Barnett, G.O., Cimino, J.J., Hupp, J.A. and Hoffer, E.P., 1987. DXplain: an evolving diagnostic decision-support system. JAMA, 258(1), pp.67-74.

[9] Warner, H.R., Haug, P., Bouhaddou, O., Lincoln, M., Warner Jr, H., Sorenson, D., Williamson, J.W. and Fan, C., 1988, November. ILIAD as an expert consultant to teach differential diagnosis. In Proceedings of the Annual Symposium on Computer Application in Medical Care (p. 371). American Medical Informatics Association.

[10] Bodenreider, O., Nelson, S.J., Hole, W.T. and Chang, H.F., 1998. Beyond synonymy: exploiting the UMLS semantics in mapping vocabularies. In Proceedings of the AMIA symposium (p. 815). American Medical Informatics Association.

[11] Rindflesch, T.C., Kilicoglu, H., Fiszman, M., Rosemblat, G. and Shin, D., 2011. Semantic MEDLINE: an advanced information management application for biomedicine. Information Services and Use, 31(1-2), pp.15-21.

[12] Swanson, D. R. "Fish oil, Raynaud's syndrome, and undiscovered public knowledge." Perspectives in biology and medicine 30.1 (1986): 7-18.

[13] Swanson, D.R. "Migraine and magnesium: eleven neglected connections." Perspectives in biology and medicine 31.4 (1988): 526-557.

[14] Smalheiser, Neil R., and Don R. Swanson. "Using ARROWSMITH: a computer-assisted approach to formulating and assessing scientific hypotheses." Computer methods and programs in biomedicine 57.3 (1998): 149-153.

[15] Swanson, Don R. "Two medical literatures that are logically but not bibliographically connected." Journal of the American Society for Information Science 38.4 (1987): 228.

[16] Ferrucci, D., Levas, A., Bagchi, S., Gondek, D. and Mueller, E.T., 2013. Watson: beyond jeopardy!. Artificial Intelligence, 199, pp.93-105.

[17] S.M. Lee, S. R. Sukumar, S. Hong and S.-H. Lim, “Enabling Graph Mining in RDF Triplestores using SPARQL for Holistic Graph-Analysis”, Expert System and Applications, Vol. 48, pp. 9-25, 2016.

[18] S.M. Lee, S- H. Lim, T.C. Brown, S. R. Sukumar, ”Graph mining meets the Semantic Web”, in the Proc. of the Data Engineering meets the Semantic Web Workshop in conjunction with International Conference on Data Engineering, April 2015.

[19] S- H. Lim, S.M. Lee, G. Ganesh, T.C. Brown and S.R. Sukumar, “Graph processing platforms at scale: practices and experiences, in Proc. of the IEEE International Symposium on Performance Analysis of Systems and Software, March 2015.

[20] S. Hong, S.-H. Lim, S. Lee, S.R. Sukumar, R. R. Vatsavai, “Benchmarking High Performance Graph Analysis Systems with Graph Mining and Pattern Matching Workloads”, in the Proc. of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (IEEE Supercomputing) 2015.

[21] S. Hong, M. Lee, S.-H Lim, S.R. Sukumar, R. Vatsavai, “Comprehensive Evaluation of Graph Pattern Matching in Graph Analysis Systems”, in the Proc. of the 25th International Symposium on High-performance Parallel and Distributed Computing (HPDC), Kyoto, Japan, 2016.

[22] Sukumar, S. R. and Keela C. Ainsworth. “Pattern search in multi-structure data: a framework for the next-generation evidence-based medicine.” *In SPIE Medical Imaging*, pp. 90390O-90390O, February 2014.

[23] Powers, S., and Sukumar, S. R. “Defining *normal* metrics for mining heterogeneous graphs at large scales”, in the Proc. of INFORMS Workshop on Data Analysis, 2014.

[24] S. Hong, S. Lee, and S-H. Lim, S.R. Sukumar, and R.R. Vatsavai., “Optimizing Graph Operations on Linked Data“, ORNL Tech Report, TM/2015/342, July 2015.

[25] L.-S Tsay, S. R. Sukumar, Larry Roberts, “Scalable Association Rule Mining with Predicates on Semantic Representations of Data”, in the Proc. of the Technology Application of Artificial Intelligence, 2015.

[26] S. R. Sukumar, N.A. Bond, K.C. Ainsworth, T.C. Brown, L. Roberts and S. Lee, “EAGLE: An App Store for Urika-GD”, in the Proc. of the Cray User Group Conference, 2015.