LarKC: The Large Knowledge Collider

EU Integrated Project FP7 - 215535
http://www.larkc.eu

Overview of the LarKC Project

Mission

The Large Knowledge Collider is a pluggable algorithmic framework for reasoning at Web-scale implemented on a distributed computational platform. It will trade quaqlity for computational cost by embracing incompleteness and unsoundness.

Instead of being built only on logic, the Large Knowledge Collider exploits a large variety of methods from other fields: cognitive science (human heuristics), economics (limited rationality and cost/benefit trade-offs), information retrieval (recall/precision trade-offs), and databases (very large datasets). These techniques are then coherently integrated within a plugin-based platform architecture.

The Large Knowledge Collider aims at an implementation on parallel hardware using cluster computing techniques, and will be engineered to scale to very large distributed settings.

Vision

The driving vision behind LarKC is to go beyond the limited storage and inference solutions currently available for semantic computing. For this purpose such an infrastructure must go beyond the current paradigms which are strictly based on logic by fusing reasoning with complementary techniques e.g. from information retrieval.

The overall vision of LarKC is to build an integrated platform for semantic computing on a scale well beyond what is currently possible. The platform aims to fulfill needs in sectors that are dependent on massive heterogeneous information sources such as telecommunication services, bio-medical research, and drug-discovery. LarKC is based on a pluggable architecture in which it is possible to exploit techniques and heuristics from diverse areas such as databases, machine learning, cognitive science, Semantic Web, and others.

Architecture

The Large Knowledge Collider architecture consists of a number of plugable components: retrieval, abstraction, selection, reasoning and deciding. These components are combined in a algorithmic schema and can be arranged in specific *workflows*. Researchers can design and experiment with multiple realisations for each of these components. Massive inference is achieved by distributing problems across heterogeneous computing resources, which are coordinated by the LarKC platform. Some of the distributed computational resources run highly coupled, high performance inference on local parallel hardware before communicating results back to the distributed computation.

```
loop
   Obtain a selection of data (IDENTIFY)
   transform to an appropriate representation (TRANSFORM)
   draw a sample (SELECTION)
   reason on the sample (REASONING)
   if more time is available
   and/or the result is not good enough (DECIDING) then
      increase the sample size (RETRIEVAL)
   else
      exit
   end if
end loop
```

LarKC allocates resources strategically and tactically, according to its basic algorithmic schema, to 1) retrieve raw content and assertions that may contribute to a solution, 2) abstract that information into the forms needed by its heterogeneous reasoning methods, 3) select the most promising approaches to try first, 4) reason, using multiple deductive, inductive, abductive, and probabilistic means to move closer to a solution given the selected methods and data, and 5) decide whether sufficiently many, sufficiently accurate solutions have been found, and, if not, whether it is worth further computation. This basic problem framework is supplied as a plug-in architecture, allowing intra-consortium and extra-consortium researchers and users to experiment with new techniques to automated reasoning.

Further Information

LarKC Website

http://www.larkc.eu

LarKC Blog

http://blog.larkc.eu

LarKC Wiki

http://wiki.larkc.eu

LarKC Platform and Plug-ins

http://larkc.sourceforge.net

LARKC LARGE KNOWLEDGE COLLIDER

LarKC: The Large Knowledge Collider
EU Integrated Project FP7 - 215535
http://www.larkc.eu

Identifying and Selecting

Introduction

Identifying and selecting in LarKC is about efficiently locating propositions contained in large-scale semantic repositories. This task is needed to support ceiling-free reasoning: we need to be able to dynamically reduce or expand the data set we are working with depending on *factors* such as *cost of processing* or *confidence* in result.

To do this we exploit methods from Information Retrieval (IR), Machine Learning (ML), and cognitive science, however in most cases the relevant methods need adaptation to the new context.

Current Status

IR models have proven their Web-level scalability and ability to generalize over contradictory data. We have applied the most straightforward derivations of this type of method (such as the minimalist techniques derived from standard Boolean retrieval models) to triple selection, relative to the above mentioned factors; this forms the baseline by which we measure progress on more advanced methods. These more advanced methods include:

- *Spreading Activation approaches, which add context-sensitivity to information retrieval*, based on biologically inspired connectionist networks associated with weighted RDF graphs; we have developed and experimented with spreading activation and PageRank components for selection, and performed initial evaluation on the Linked Data Semantic Repository (LDSR) dataset.
- *Spreading Activation approaches inspired by cognitive models* of human memory: We have studied the relationship between the network structure of human memory and knowledge bases; we consider RDF data sets as a network, and use centrality measures borrowed from network statistics as criteria for selection. For that purpose we performed initial experiments on the IMDB dataset.
- Moreover, *cognitive memory retention-like models* from interest retention extraction and search refinement have been applied to the SwetoDBLP dataset. A DBLP interest enhanced dataset has been released.

Future Challenges

In the future, we plan to improve existing methods for selection in order to support large datasets such as those provided by use cases within LarKC related to cancer research and drug developement. More precisely, our remaining challenges are:

- To improve existing methods based on the evaluation driven by the use-case in which we can test accuracy and performance. Although our main challenge is to focus on improving and developing selection methods, another big challenge is the proper evaluation of such methods due to the nature of RDF and the goal of the selection process itself;
- To explore, apply and evaluate more advanced techniques from IR such as Random Indexing;
- To explore, apply and evaluate Machine Learning (ML) methods for selection;
- Develop LarKC plugins for those methods that prove to be successful in experimentation;
- Experiment with running selection components on a parallel platform.

Further Information

Identifying and Selecting within LarKC

http://wiki.larkc.eu/LarkcProject/WP2

Linked Data Semantic Repository

http://www.ontotext.com/ldsr

Linked Life Data

http://www.linkedlifedata.com

IMBD Dataset

http://www.imdb.com

http://wiki.larkc.eu/csri-rdf

LarKC Large Knowledge Collider

LarKC: The Large Knowledge Collider
EU Integrated Project FP7 - 215535
http://www.larkc.eu

Transforming

Introduction

Transformation refers to converting and extracting information from various sources and formats to a representation that is suitable for further processing within the LarKC platform. Two main efforts towards this are Machine Learning and work on integrating data streams.

Machine Learning efforts are based on the assumption that not all regularities in a domain can be captured by a logic-based ontology but rather require a statistical analysis. Recommendation systems, for example, extract preference patterns from data, e.g. *which kind of users may likely want to watch movies of which genres?* While Machine Learning is not new, we want to address new specific challenges within LarKC. In particular, it should be possible to integrate machine learning predictions into querying. For the user there should be no difference between querying facts, deduced facts and induced information. For that purpose we are extending SPARQL, the Semantic Web query language, to allow the integration of learned probabilistic statements into querying. Furthermore, Machine Learning should be able to deal with the typical data situation of the Semantic Web with highly sparse information, missing information and extremely large scale.

Data Streaming concerns a type of data common in many real world applications (e.g. monitoring of network traffic, telecommunications management, clickstreams, manufacturing, sensor networks, etc.) but not well treated within the Semantic Web. A first step toward Management of Semantic Streams is to combine the power of existing Data Stream Management Systems (DSMS) and Semantic Web technologies.

Current Status

To demonstrate the effectiveness of Machine Learning a demonstration prototype is under development which operates as follows: A first step is the construction of a learning matrix which consists of features (columns) associated with the key entities (rows) in the sample. The features are derived from the triples involving key entities and from aggregated information. In the learning procedure, the truth values of unknown triples are estimated and can be written into the knowledge base. A key issue in this process is the representation of different certainty values for the learned triples. Our demonstrator prototype shows how learned triples with their certainty value can be integrated into extended SPARQL queries.
In the work on Data Streaming, the notion of RDF streams as the natural extension of the RDF data model was introduced. An RDF stream is defined as an ordered sequence of pairs, where each pair is constituted by an RDF triple and its timestamp. This definition of RDF streams extends RDF in the same way as the stream type in DSMS extends the relation type. Furthermore, we have introduced Continuous-SPARQL (or simply C-SPARQL) as an extension of SPARQL for querying both RDF graphs and RDF streams. The distinguishing feature of C-SPARQL is its support for continuous queries, i.e. queries that are registered and then executed continuously over windows opened on RDF streams and standard RDF graphs.

Future Challenges

After having demonstrated all basic steps in the machine learning approach, our current goal is to realize an easy-to-use statistical Machine Learning plug-in to predict relationships in semantic domains of interest (e.g. Linked Data). The main technical issues to be addressed are proof of scalability in terms of computational requirements and storage requirements, sampling issues, benefits from active learning, prediction accuracy and usability. Furthermore, we will demonstrate the wide applicability of Machine Learning techniques within the LarKC use cases (e.g. recurrent neural networks applied for predicting traffic speed in urban computing use case) and beyond.

Regarding data streaming solutions, we aim at developing a C-SPARQL engine, deploying it as a transformation plug-in in the LarKC platform and demonstrating its use on real world RDF streams. Beyond that we will investigate query rewriting techniques to leverage existing Data Stream Management Systems, continuous query plan adaptation to the bursty nature of data streams, cost metrics to measure query plan cost, and intra and inter stream parallelisation.

Further Information

Materializing and Querying Learned Knowledge

http://www.larkc.eu/shortlink/5

C-SPARQL in LarKC

http://wiki.larkc.eu/c-sparql

C-SPARQL Demo

http://c-sparql.cefriel.it/sdow-demo/

LarKC Large Knowledge Collider

LarKC: The Large Knowledge Collider
EU Integrated Project FP7 - 215535
http://www.larkc.eu

Reasoning and Deciding

Introduction

The essence of the LarKC project is to go beyond notions of absolute correctness and completeness in reasoning. We are looking for retrieval methods that provide useful responses at a feasible cost of information acquisition and processing. Therefore, generic inference methods need to be extended to capture this notion of a trade-off between completeness/correctness and scalability.

In LarKC we do not limit ourselves to any specific reasoning technique. Approximate reasoning is a non-standard reasoning approach based on the idea of sacrificing soundness or completeness for a significant speed-up in reasoning. This is done in such a way that the loss of correctness is at least outweighed by the obtained speed-up. Parallel reasoning and distributed reasoning are considered to be essential for Web-scale reasoning to improve scalability. Stream reasoning provides the reasoning support in which memory overload is avoided by operating on streams of data instead of statically available sets. Granular reasoning is a non-standard reasoning approach in which multiple perspectives/views can be selected for reasoning by using knowledge at various levels of specificity and data at variable levels of granularity.

We aim to construct several reasoning plug-ins, based on insights from both generic inferencing methods and non-standard reasoning, and invite third parties to contribute further components to the LarKC eco-system. Inspired by work on meta-reasoning, the reasoning process and the interaction between different components is in turn coordinated by workflow-determining *decider* plug-ins.

Current Status

So far we have developed several reasoner plug-ins for the LarKC platform. We have integrated several Description Logic reasoners, such as Pellet, Racer, FaCT++, KAON2, etc. into the LarKC platform. Additionally, we developed a rule-based inference plug-in for the LarKC platform, which is based on the IRIS datalog engine. IRIS can be configured with different rule-sets and can be used to compute inferences for RDF(S) and several subsets (profiles) of OWL.
Work on granular reasoning has shown an initial capability for interleaving reasoning and selection of sub-sets of data with an emphasis on providing reasoning results according to different user backgrounds and perspectives and different requirements for certainty and specificity. Furthermore, research results towards new reasoning methods have resulted in experimental approaches for mapping human search strategies. This research will link biological models of animal foraging with psychological theories on heuristic problem solving, stopping rules and task switching. Based on this we will further analyse human decision processes in the context of memory retrieval and problem solving.
Both manually configured (or scripted) deciders and and automated decider using formal plug-in descriptions are currently deployable on the LarKC platform. The automated deciders combine plug-ins according to their functionality by using the Cyc knowledge-base and inference engine to reason over plug-in descriptions.

Future Challenges

One of the main challenges to be addressed within LarKC is reasoning at a Web scale. In particular the main characteristics identified in this regard are as follows:

- **Infiniteness**. There already exists an exceedingly large number of data on the Web. By June 2009, the Linking Open Data initiative amounted to more than four billion triples and is expected to grow rapidly.
- **Dynamics**. Web scale data is in flux. Thus, we cannot assume static and monolithic knowledge bases but rather have to expect constant updates and changing data.
- **Inconsistency**. Knowledge on the Web is bound to be of widely varying quality and to express several different viewpoints. Re-using and combining multiple data-sets found on the Web is thus bound to lead to inconsistencies between the combined vocabularies. According to classical entailment, contradictions entail any proposition. Within LarKC we aim at novel reasoning methods that prevent this explosion and manage to perform meaningful inference in the presence of inconsistencies.

Further Information

A Survey of Web Scale Reasoning

http://www.larkc.eu/shortlink/4

Strategies and Design for Interleaving Reasoning and Selection of Axioms

http://www.larkc.eu/shortlink/3

LarKC Large Knowledge Collider

LarKC: The Large Knowledge Collider
EU Integrated Project FP7 - 215535
http://www.larkc.eu

The LarKC Data Layer

Introduction

The *data layer* is a core platform component that supports all plug-ins and applications with respect to storage, retrieval, and light-weight inference on top of large volumes of data. The layer implements a range of data access interfaces including a Java API or a SPARQL endpoint.
The main features and functionalities that the Data Layer provides to the rest of the LarKC platform are:

- Persistent storage of RDF data,
- Passing RDF data by value or reference,
- Resolvable RDF data identifiers,
- Efficient SPARQL query optimisation and execution.

Current Status

The default data layer implementation is based on the Ontotext's OWLIM engine. Detailed benchmark results of various versions of the data layer can be found in the OWLIM Benchmarking section of the Ontotext web site. The results can be summarized as follows:

- The data layer can perform lightweight reasoning against 12 billion explicit statements; it loads a dataset of LUBM(90,000) at an average speed of 11,5 KSt/sec;
- The data layer can deal with 20 billion statements on a single server worth less than $10,000. This is the total number of statements inserted in the indices after materialization of LUBM(90k). Although direct loading of the same number of statements without reasoning would be faster, this results give a good estimate concerning the scalability of the data layer.
- The data layer can deal with 1 billion statements on a desktop machine worth $2,000 and it takes less than 5 hours to load the dataset; loading time including inference and materialization amounts to 14 hours; The full-cycle run of LUBM(8000), including loading, inference, and query evaluation, takes 15.2 hours.

The Linked Data Semantic Repository (LDSR) was used to test the data layer with a real-world dataset. LDSR consists of more than one hundred thousand different predicates, plenty of literals of various size, and long chains of transitive properties.

Dataset	Explicit Indexed Triples ('000)	Inferred Indexed Triples ('000)
DBPedia (SKOS categories)	2,233	262,734
DBpedia (owl:sameAs)	2,053	4,006
UMBEL	3,197	41,228
Lingvoj	20	112
CIA Factbook	161	40
Wordnet	1,943	5,236
Geonames	72,749	471,220
DBpedia 3.3 core	357,450	360,172
Total	439,815	1,144,755

Future Challenges

The current LarKC data layer provides the functionality for requesting large sets of RDF triples from a central repository. These features already guarantee a certain level of scalability and performance in cases where few requests have to be processed concurrently. Several different approaches are under investigation to increase its scalability.
In particular, we aim at a Distributed Data Management infrastructure, based on P2P technologies. It will address the issues of multiple concurrent requests, management of extremely large sets of RDF triples and access to multiple and distributed data sources, ensuring, at the same time, interoperability with the current data layer (implementation and API), by development of a set of data management services to maintain connectivity with peers, coordinate (complex) queries and manage access to distributed data sets (replication, partitioning).
Futhermore, we envision the development of a set of metadata/context management services, which provide additional information by using a virtual overlay network, in order to improve performance of searches and quality of results.

Further Information

Linked Data Semantic Repository

http://www.ontotext.com/ldsr/

OWLIM Benchmark Results

http://www.ontotext.com/owlim/benchmarking/index.html

LarKC Large Knowledge Collider

LarKC: The Large Knowledge Collider
EU Integrated Project FP7 - 215535
http://www.larkc.eu

LarKC Use Cases

Urban Computing

Urban Computing is a branch of Pervasive Computing that investigates urban settings and everyday lifestyles. Pervasive Computing has been studying glaciers, coral reefs, and smart rooms by deploying sensor networks and analyzing the collected data. However, Pervasive Computing largely neglected urban settings.However, the information to develop pervasive applications for urban environment is often already available: maps, points of interest, user locations, traffic, pollution, and events are just a few examples of the digitalized information which we can access on the Web. The LarKC platform is allowing us to put together such information and to provide a better answer to everyday questions such as: "Which point of interest can I visit from here?" "Can I get there quick enough considering the current traffic situation?" The current state of our work is documented in the Alpha Urban LarKC movie as well as in a prototype that is available online.

Early Clinical Development

Early clinical development is a bottleneck within drug development; examining the applicability of findings in vitro and in animals to humans is a huge challenge. The complexity in drug development is out-stripping the abilities of individual humans or even teams of humans. Therefore, the LarKC platform will be used as a tool to integrate and interpret a massive number of heterogeneous data sources. We have developed four case stories that present the need of a holistic view to existing biomedical knowledge. Scientists at AstraZeneca R&D evaluate and guide the use case implementation.

- The first case, the epidemiology story, addresses the need for long term knowledge building for disease and patient stratification. The underlying knowledge about diseases and different phenotypes (patient groups) have big gaps and with the LarKC and the Linked Life Data (LLD) repository we will aim to fill some of them.
- The second case, concerns the "project memory": A project team in early clinical development includes many scientific disciplines and members from different parts of the organization. To give them a common view of the project knowledge and to bridge the different data domains we aim to use LarKC and LLD as a project memory.
- The third case, the data interpretation story, aims to assist the biomarker/proteomics team with the interpretation of data.
- The fourth case, the safety evaluation case, addresses the need to understand the case of safety events. The aim is to use LarKC as a complement approach when evaluating adverse event reports for drugs.

Carcinogenesis Research

The LarKC platform will be used to assist carcinogenesis researchers in two scenarios. In the first scenario, LarKC will assist in the production of standard reference publications on carcinogens. These *monographs* are definitive evaluations of all published research on a carcinogen. Given the large amount published on cancer, finding and sifting this literature is an enormous task. LarKC will provide novel ways of searching and navigating the literature.
In the second scenario, LarKC will assist in studies of the association between genes and cancer. These studies are complicated by small sample numbers and large numbers of variables. A gene must have an extremely strong association with a disease to appear significant in these studies. LarKC will help to include background knowledge on associations by bringing to bear the literature from previous research in the area.
In both scenarios, semantic annotations will be used to find the relevant concepts in the biomedical literature: finding genes, proteins, disease, research scientists, etc. mentioned in research articles. Linking these annotations to large life science data repositories will provide a rich network of knowledge. The LarKC platform and its components will be used to process this knowledge, presenting carcinogenesis researchers with links and relationships that they had not previously considered.

Further Information

LarKC Use Case: Urban Computing

http://wiki.larkc.eu/UrbanComputing/AlphaUrbanLarkc

Alpha Urban LarKC Online

http://www.larkc.eu/shortlink/6

Alpha Urban LarKC Demo Movie

http://www.larkc.eu/shortlink/7

LarKC Use Case: Early Clinical Development

http://www.larkc.eu/shortlink/1

LarKC Use Case: Carcinogenesis Research

http://www.larkc.eu/shortlink/2

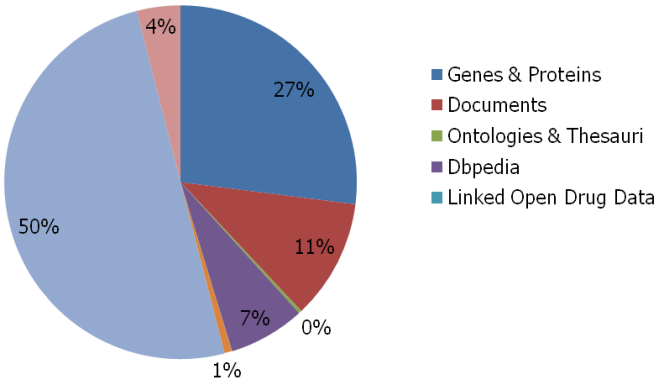
LarKC Large Knowledge Collider

LarKC: The Large Knowledge Collider
EU Integrated Project FP7 - 215535
http://www.larkc.eu

LarKC Data Sets

Linked Life Data

Data integration continues to be a serious bottleneck for the expectations of increased productivity in the pharmaceutical and biotechnology domain.
The 2009 annual database issue of "Nucleic Acid Research" cited 1170 existing molecular biological data sources. Linked Life Data integrates common public datasets that describe the relationships between gene, protein, interaction, pathway, target, drug, disease and patient. The existing structured knowledge is enriched with data from the PubMed database which comprises more than 19 million citations for biomedical articles from MEDLINE and life science journals. Furthermore, LLD also includes data from the UMLS metha-thesaurus, which semantically integrates many controlled vocabularies in the biomedical sciences and ontologies. The Linked Life Data datasets consist of more than 5 billion RDF statements.



The dataset interconnects more than 20 complete data sources and helps to understand the "bigger picture" of a research problem by linking previously unrelated data from heterogeneous knowledge domains. To make efficient usage of the public linked data cloud we have created instance alignment patterns that restore missing information relationships. As a final step, a great many semantic annotations (optimized for high recall or precision) are generated, linking semantic data instances and the unstructured information.

Milan Municipality Data Set

Urban environments are represented on the Web through a large and diverse set of distributed pieces of information: maps, events, interesting places, traffic data, etc. Moreover, local governments' awareness of publishing data on the Web for public use is increasing. Concentrating our attention on the city of Milano in Italy, we have identified several data sources to take into account for our Urban Computing scenario. These data sources are diverse and heterogeneous not only in content, but also in format and availability conditions. For those reasons, we not only gathered data, but we also elaborated, processed and, in some cases, converted them to RDF as an interchange format.
In particular we currently include the following data sources:

- **Traffic Sensor Data from Milano:** This datasource contains information about speed, flow, and occupation rate from stationary sensors on the roads of Milano. In addition, the data set contains information about whether a certain day was a holiday and whether it was in a low traffic season, descriptions of the different vehicle classes, and descriptions of the different time slots (5min intervals). This data set amounts to 10^{10} triples if completely converted to RDF and is also very challenging with respect to the quality of the data.
- **lMeteo.it Historical Weather Data:** This data source provides weather data for all Italian municipalities. ilMeteo.it offers weather predictions every day, but it also offers historical data of the weather measurements of past years, ranging from 1973 to today. Converted to RDF the archive amounts to 10^9 statements. This weather data is very useful for traffic prediction since its integration with traffic information can give useful insight on the causes of traffic congestions.
- **Glue social network interactions:** Furthermore, we include data provided by Glue as as a data stream. Glue enables users to connect with their friends on the Web based on the pages the users visit online. Using semantic recognition technologies it generates a continuous stream which is accessible in real time through a REST API. Stream data collected between 3.8.2009 and 4.9.2009 results in a data set that, if stored completely, would amount to 2.768.434 triples.

Further Information

Urban Computing in LarKC

http://wiki.larkc.eu/UrbanComputing/

Linked Life Data website

http://linkedlifedata.com

Glue

http://getglue.com/

LarKC Large Knowledge Collider