The DILIGENT Framework for Distributed Information Retrieval

Fabio Simeoni
University of Strathclyde
Glasgow, UK
f.simeoni@cis.strath.ac.uk

Fabio Crestani
University of Lugano
Lugano, Switzerland
fabio.crestani@unisi.ch

Ralf Bierig
University of Strathclyde
Glasgow, UK
r.bierig@cis.strath.ac.uk

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software

General Terms

Algorithms

Keywords

GRID, Service Oriented Architecture, Distributed Information Retrieval, Digital Libraries

1. INTRODUCTION

To date, the adoption of content-based Distributed Information Retrieval (DIR) solutions in the practice of information services remains rather limited. While state-of-the-art DIR prototypes have been recently announced [1], real-world applications are largely confined to Web meta-searchers and appear to make little use of the rich array of techniques reported in the literature. We argue that a major obstacle to the uptake of DIR solutions is the lack of a development and deployment infrastructure built around *open application frameworks* and *standard formats, languages, and protocols.* Developments outside the DIR field have rallied some consensus around query protocols[1], but true application frameworks for DIR remain unavailable and developers are forced to either home-grow solutions or to repurpose simulation code intended for ad-hoc research evaluations[2]. In this paper, we report on the ongoing design and implementation of an application framework for DIR.

2. THE DILIGENT FRAMEWORK

The *DILIGENT framework* for DIR takes its name from the ongoing EU project which funds its development and defines its initial context of application: namely, a fully distributed infrastructure of middleware and application services built for the Digital Library (DL) domain upon the lower-level facilities of a European GRID platform[3]. The DILIGENT goal is to lift the GRID philosophy of dynamic and coordinated resource sharing into the DL domain by allowing the definition of virtual DLs which are: (*i*) built declaratively from community-provided datasets and *application services*, and (*ii*) deployed *on demand* by *middleware services* across machines and according to availability, performance, and functional constraints[4].

Within the DILIGENT infrastructure, the DIR framework supports the development of DIR application services: namely, *Content Source Selection* (CSS), *Content Source Description* (CSD), and *Data Fusion* services (DF) [4][5]. In particular: (*i*) CSS services limit the routing of queries to the sources which appear to be the best targets for their execution, where 'goodness' criteria include the relevance of content, the sophistication of retrieval engines, and or the monetary costs associated with query execution; (*ii*) CSD services generate and maintain summary descriptions of content sources, such as partial indices, collection-level term statistics, or result traces from training or past queries; (*iii*) DF services derive a total order of the result lists produced autonomously by target sources,e.g. by normalising with respect to different scoring functions and content statistics.

The framework aims at supporting a wide range of strategies across the three core tasks of DIR. For example, a CSD service may base source descriptions on term statistics derived from full-text content indices, while another may do so using partial indices derived directly from the content through query-based sampling techniques [2]. Similarly, a DF service may rely on a round-robin algorithm to merge results which emanate from different sources, another may be biased by the output of a CSS service, and yet another may employ non-heuristic techniques and leverage the output of a CSD service towards forms of 'consistent' fusion.

Though pairwise different, strategies of description, selections, and fusion share a common architecture which the framework hopes to capture into extensible state-of-the-art components. In particular, the design of the framework has been informed by the following requirements: (*i*) (*flexibility*) it should support the implementation and configuration of the wide range of cooperative and uncooperative strategies which characterise the current scope of the DIR field; (*ii*) (*efficiency*) it should optimise the consumption of computational resources, including computing cycles, bandwidth and storage; (*iii*) (*responsiveness*) it should minimise the latencies and interruptions of service typically associated with

[1] See `http://www.loc.gov/standards/sru/srw`.

[2] See `http://www.lemurproject.org.`

[3] See. `http://www.eu-egee.org.`

[4] See. `http://www.diligentproject.org.`

[5] Within DILIGENT, search distribution and content indexing functionality are provided by separate services and will not be discussed here.

SIGIR'07, July 23–27, 2007, Amsterdam, The Netherlands.
ACM 978-1-59593-597-7/07/0007.

DIR processes which make real-time use of the network; (*iv*) (*openness*) it should adopt open technology standards, from data formats and applications protocols, to language and systems platforms.

Approaching its beta version at the time of writing, the framework offers transparent interaction with the infrastructure (e.g. best-effort publication and discovery of services; modelling, persistence, bootstrapping, and recovery of state; GRID-based file exchange mechanisms; cross-service subscriptions and notification mechanisms) and an extensible pattern-based design to support DIR-specific functionality (e.g. local inverted indexing for selection and fusion indices, by-reference data exchange for streamed input and output of result lists, update policies for source descriptions).

3. REFERENCE SERVICES

We have tested the framework by developing an initial set of *reference services* for the DILIGENT testbed.

The reference CSD service generates and maintains term histograms of textual sources, a coarse-grained form of index where containment relationships between terms and documents is intentionally abstracted over. The service interacts with DILIGENT Index services to derive the histograms from full-text indices of collections and also to subscribe for point-to-point notifications of changes to such indices. The histograms are exposed to clients via synchronous calls suitable for fine-grained access, but also through asynchronous, file-based exchange suitable for coarse-grained access. The regeneration of histograms occurs with respect to update policies based on a configurable combination of time and space criteria (i.e. every so often and/or whenever the index has changed of a given proportion).

The reference CSS service selects sources based on rankings produced either with the standard CORI algorithm [3]. Invoked by the DILIGENT Search service during query execution, the service can be configured to select the sources below a given cut-off point in the ranking, or else to derive such cut-off from an upper bound on the number of results to be retrieved; in the latter case, it returns an indication of the number of documents to retrieve from selected sources. Rankings are based on estimated relevance of content and rely on term histograms staged from the reference CSD service prior to query submission. In particular, the service subscribes with the reference CSD service for changes to the staged histograms and updates them upon receiving notification of such changes.

Finally, the reference DF service merges query results based on either one of three techniques: a plain round-robin algorithm, a consistent merging algorithm, and a linear regression method based on source selection scores. The first offers the least effectiveness but acts as an upper bound on performance (results remain unparsed, output can be streamed). The second uses global statistics to give the best effectiveness but also the highest overhead (results are fully parsed, output cannot be streamed); in this case, the service interacts with the reference CSD service to gather histograms in advance of result submission. The third explores middle ground between the first two, and uses the output of the reference CSS service to heuristically normalise inconsistent result scores; as interaction with the CSS service must necessarily occur during query execution, DILIGENT middleware services can be instructed to co-deploy both services on the same node.

4. FUTURE WORK

The reference DIR services are currently employed in support of content-based retrieval over the literary texts of ARTE, a DILIGENT community led by the Scuola Normale Superiore (SNS) di Pisa and Radio Televisione Italiana (RAI) the national Italian broadcasting agency. While the functionality of the reference services is not raising particular issues, the performance of the beta version of the infrastructure as a whole needs finer tuning before entering its production phase. Accordingly, we are engaging in further optimisation of the framework code, mostly in the attempt to reduce the network latencies which characterise the DIR tasks.

In terms of functionality, we plan to extend the framework and add support for the implementation of uncooperative strategies. In particular, we plan to implement query-based sampling techniques for generating term histograms or partial indices of sources which live outside the infrastructure but interface it through query wrapping services available in DILIGENT. Partial indices, in particular, would allow us to support most advanced techniques of selection and fusion, such as the Semisupervised Learning method of data fusion (SSL)[5] and the Unified Utility Maximization method of resource selection (UUM)[6].

5. REFERENCES

[1] Thi Truong Avrahami, Lawrence Yau, Luo Si, and Jamie Callan. The fedlemur project: Federated search in the real world. *J. Am. Soc. Inf. Sci. Technol.*, 57(3):347–358, 2006.

[2] J. Callan, A. L. Powell, J. C. French, and M. Connell. The effects of query-based sampling on automatic database selection algorithms. Technical Report CMU-LTI-00-162, Language Technologies Institute, School of Computer Science, Carnegie Mellon University, 1999.

[3] James P. Callan, Zhihong Lu, and W. Bruce Croft. Searching distributed collections with inference networks. In *SIGIR '95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 21–28, New York, NY, USA, 1995. ACM Press.

[4] Nicholas Eric Craswell. *Methods for Distributed Information Retrieval.* PhD thesis, Australian National University, May 2000.

[5] Luo Si and Jamie Callan. A semisupervised learning method to merge search engine results. *ACM Trans. Inf. Syst.*, 21(4):457–491, 2003.

[6] Luo Si and Jamie Callan. Unified utility maximization framework for resource selection. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 32–41, New York, NY, USA, 2004. ACM Press.