1 Introduction

In the recent years, with the increasing prevalence of databases online, the Web has been rapidly “deepened.” Numerous sites now provide access to their contents through query interfaces. For example, to obtain the used car listings on cars.com, users would query by make, model, etc. Some surveys [1, 2] estimated about 25 million deep Web sources. Building novel integration applications, e.g., kayak.com in airline, and trulia.com in real-estate domain, has clearly become the next frontier of information access on the Web.

Inspite of the large and ever expanding scale, full of promising opportunities, there is no effective exploration facility for the deep Web. Such a facility will be important for building large scale integration systems in any domain. As an example, consider Amy, an integrator for book domains. She would like to find alternative sources to amazon.com or bn.com. Thus, any integration starts with finding “where and what sources are?” Further, integrators also need to know “what vocabulary sources commonly speak?” For the airline domain, Sue will need to know that the attributes from to, departure date, and return date are common across airline sources. Thus, for finding related sources and bringing them together, integrators need to know:

- T1: What are sources that are alternative to amazon.com? Related to aa.com?
- T2: What are sources about “books”? Or “flights”? What are sources that can answer queries like “make = GM”?
- T3: What are common queries used by amazon.com and albooks.com? What are their common keywords?
- T4: What are common attributes used with title and author? Or by sources about “books, fictions”?

In this demo, we present Dewex\(^1\), an exploration facility for answering questions like T1 - T4, running on repository of over 30,000 real sources. The main technical contributions of this demo are:

Techniques and Contributions:
- As our system, we present a novel facility for exploring the deep Web.

\(^1\)DEep Web EXplorer

Figure 1: Dewex: An exploration facility for deep Web.

- As our solution, we propose a schematic metadata based source modeling, and an generalized search mechanism to compute associativity in metadata graph.
- In our realization, to enable interactive online exploration, we propose to speed up computation using using matrix optimization.

2 Design of Exploration System

Source Model - Schematic Metadata: We model a source using its query attribute, describing the schema of the data and representative keywords, describing the functionality of the source. These metadata elements can be easily extracted from the interface pages of these sources.

Exploration Mechanism - Associativity Search: Schematic metadata modeling offers opportunities to determine the underlying associativity. A particular metadata element would tend to occur in multiple related sources. The associativity of source and metadata can thus be captured using the co-occurrence, which can evolve into higher order transitive co-occurrence.

To measure the associativity via occurrences of various orders, we use a graph-based representation of source modeling. With such a graph-based representation, we quantitatively define associativity measure so as to model our problem as a Markov chain based link analysis. In interest of space, we refer the interested readers to our technical report [3] for more details.

Online Computation - Matrix Optimization: To provide an interactive exploration, we need efficient optimization to speed up the computation. As the nature of the computation is matrix manipulation we develop “spatial” decomposition (exploring matrix localities) and “temporal” aggregation (exploring matrix precomputation).
In the “online” stage, as input, our exploration system accepts queries containing combination of sources, attribute and keywords. Given such a query, it produces a ranked output for all types of nodes.

In its “Offline” phase, as shown in Figure 2 system begins with collecting the sources from deep Web using the database crawler. It applies attribute extractor and keyword extractor to obtain the schematic metadata for constructing the metadata graph. To apply a link analysis on the metadata graph, the indexer builds column-major-index, which are essentially inverted index lists to each column of the propagation matrices. To speed up the online computation, matrix optimization applies the spatial and temporal matrix optimizations to prepare the indexes used in online computation.

3 Demo Description

With system integrators as our target users, we demonstrate the utility of Dewex by showcasing example scenarios for tasks $T_1$ - $T_4$:

**Task T1 (source $\rightarrow$ source):** Consider airline domain, where integrator wants to find sources like aa.com. In Figure 3 (a), we show the top sources for the query, $Q_1 = \{aa.com\}$.

**Task T2 (metadata $\rightarrow$ source):** For the same requirement, a different developer, gives a keyword query, $Q_2 = \{flight, airfare\}$. The top results, as listed in Figure 3 (b), are indeed all in airline domain.

**Task T3 (source $\rightarrow$ source):** Now consider an integrator trying to find attributes commonly used in book domain. For her query, $Q_3 = \{amazon.com, a1books.com\}$, the top attributes, as listed in Figure 3 (c), are all very intuitive.

**Task T4 (metadata $\rightarrow$ metadata):** For the same application, another integrator, simply specifies some keywords in her query, $Q_4 = \{fiction, bookstore\}$. The top ranked attributes for $Q_4$ are listed in Figure 3 (d).

**User Interaction:** As a front end to our system, we provide a web based interface, as shown in Figure 4, where users can give input queries. Our simple interface contains only a single keyword box, which can be used to to submit query $Q_2$. As result, ranked list of sources is presented to the user. The snippet shows a summary of the attributes and representative keywords used to model that source, and also provide hyperlinks to obtain detailed information.

Next to each snippet, there are checkboxes. A user can select some of the entries to further refine their query. They can also use them to issue fresh queries by resetting the form, and, for example, choosing aa.com to issue query $Q_1$.

In our advanced form, users can specify a mixture of various attributes, keywords and sources, such as, query $Q_3$. The links “top attributes” and “top keywords” in our result page will display the top ranked attributes and keywords.

Once again, user could modify her query by choosing any of the listed attributes or keywords, as we described earlier for sources. By choosing the keywords fiction, and bookstore, a user could issue the query $Q_4$ discussed above.

In our demo, we will present Dewex in an interactive setting to demonstrate its exploration utility in finding "where and what sources are?" and what ‘vocabulary’ sources commonly speak?

References


