MetaQuerier over the Deep Web: Shallow Integration across Holistic Sources*

Kevin Chen-Chuan Chang, Bin He, and Zhen Zhang

Computer Science Department
University of Illinois at Urbana-Champaign
{kcchang, binhe, zhang2}@uiuc.edu

Position Statement

Abstract

The Web has been rapidly "deepened" by myriad searchable databases online. To enable effective access to the "deep Web," we are building the *MetaQuerier*– for exploring and integrating databases on the Web. Such metaquerying must tackle integration at a *large scale* (as sources are proliferating online) and of a *dynamic* nature (as each query will access different sources). Toward such integration, our approach hinges on the insight that the challenge of large scale is itself an opportunity: We observe that the desired "semantics" often connects to surface presentation characteristics, through some hidden regularities over many sources. Generalizing our recent works, this paper thus proposes our approach of *shallow integration* across *holistic sources* – to discover desired semantics by exploiting the hidden regularities of shallow clues across many sources holistically. As evidences, we have studied two concrete problems: 1) query-interface understanding based on hidden syntax, and 2) query-interface matching based on hidden statistic models. Our experience indicates high promise for employing shallow techniques across holistic sources.

1 Introduction

In the recent years, the Web has been rapidly deepened with the prevalence of databases online. As Figure 1 conceptually illustrates, on this so-called deep Web, numerous online databases provide dynamic query-based data access through their *query interfaces*, instead of static URL links. A July 2000 survey [2] estimated that 96,000 search sites and 550 billion content pages in the deep Web. Our recent study [4] by random IP sampling in December 2002 estimated between 127,000 to 330,000 deep Web sources.

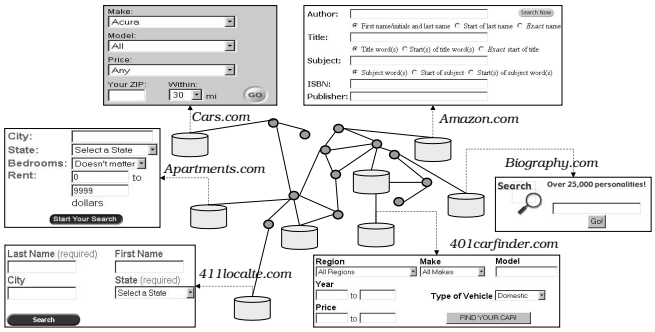


Figure 1: The deep Web.

To enable access to the deep Web, we propose to build a metaquery system, *MetaQuerier*,[1] which aims at helping users to *find* and *query* online sources, as Figure 2 illustrates. For instance, consider user Amy, who is moving to a new town. First, different queries need different sources to answer: Where can she look for real estate listings? (*e.g.*, *realtor.com*.) Studying for a new car? (*cars.com*.) Looking for a job? (*monster.com*.) Further, different sources support different query capabilities: After source hunting, Amy must then learn the grueling details of querying each source. The overall goal[2] of the MetaQuerier is thus to explore (*i.e.*, find) and integrate (*i.e.*, query) the myriad databases on the Web.

Such metaquerying faces new challenges: First, it must deal with *large scale*, since sources are proliferating rapidly online. Second, it must be *dynamic* and *ad-hoc*: each query will dynamically select different ad-hoc sources (*e.g.*, consider Amy's three queries). While tantalized by the need for effectively accessing the deep Web, such large-scale metaquerying has largely remained unexplored. As Section 2 will discuss, on one hand, research on large scale "metasearch" has focused on text sources. On the other hand, database integration efforts have mostly assumed rel-

*This material is based upon work partially supported by NSF Grants IIS-0133199 and IIS-0313260. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the funding agencies.

[1] For more details: http://metaquerier.cs.uiuc.edu

[2] In particular, the current scope (*e.g.*, as in [7]) of the MetaQuerier focuses on integrating sources in the same domain (*e.g.*, Books, Airfares).

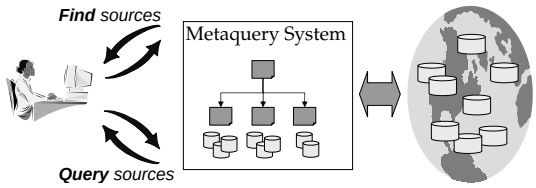


Figure 2:MetaQuerier for accessing the deep Web.

atively small scale, pre-configured sources.

In particular, the large scale and dynamic nature calls for techniques for *on-the-fly* integration: With the sheer scale of the deep Web, it is probably not feasible to acquire manually annotated knowledge (*e.g.*, source query capabilities) for each source. Further, as sources are dynamically selected to satisfy user queries, it is unlikely to rely on some pre-configured sources for ad-hoc needs.

Such on-the-fly integration, with the lack of a priori source and query knowledge, often faces challenges in discovering source "semantics"– or, the knowledge required for interacting with sources. The challenge of "semantics" discovery for integration is everywhere: At various steps of integration, different tasks will require certain aspect of such source semantics. To name a few, for the MetaQuerier to find and query online sources, there are several critical tasks that require discovering various semantics:

T1. **How to understand a query interface?** Given a query interface for a source, what are the query conditions and their attributes? This task is to extract the *query capabilities* of a source on the fly, for modeling the source– a critical first task for integrating Web sources. The "semantics" to be discovered is thus to understand what a query form says as its query conditions (*i.e.*, attributes, operators, and domain values)– *e.g.*, as Figure 7(a) shows, *amazon.com* supports several conditions, among which the first one is on author (as the attribute), with any text (as the domain values), in three name formats (the operators).

T2. **How to match query interfaces?** Given several query interfaces (say, for Books sources), what does author condition on one source match another? This task is to match query conditions across different sources. The "semantics" to be discovered is thus the synonym correspondences of attributes– *e.g.*, author of one source corresponds to name of another, and similarly subject to category.

T3. **How to translate queries across sources?** Given a query expressed on one source (or some unified interface), how to ask this query on another? This task is to rewrite a query in terms of what the interface of a target source can express. The "semantics" to be discovered is thus the closest mapping to the target query interface (*i.e.*, how to fill the query form of the target source)– *e.g.*, the query [lastname = `"Gray"`] expressible on one interface may map to [author contains `"Gray"`] on another.

As our critical insight, toward building the MetaQuerier over the deep Web, while the large scale presents new challenges, we believe it also reveals itself as novel opportunities. To explore this insight, we propose a new philosophy

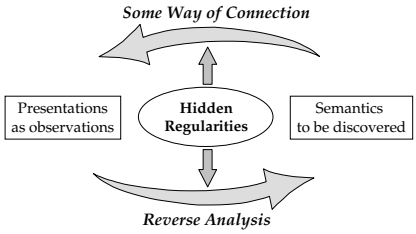


Figure 3: Shallow integration across holistic sources.

as a general approach for integration at large: that of *shallow* integration across *holistic* sources, as Figure 3 conceptually shows. Consider an integration task that requires discovery of semantics (*e.g.*, tasks T1, T2, T3). To begin with, our philosophy builds upon two hypotheses: First, *shallow observable clues*: The desired "underlying" semantics often connects (Figure 3, top) to the "observable" presentations, or shallow clues. Second, *holistic hidden regularities*: Such connections often follow some implicit properties, or hidden regularities (Figure 3, middle), which will reveal holistically across many sources.

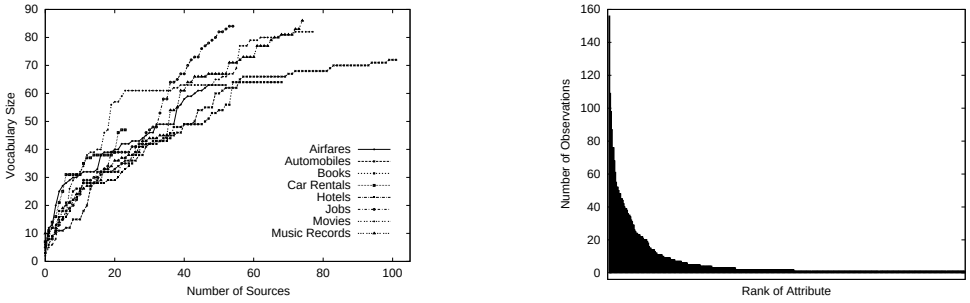
Therefore, our integration task, or the discovery of the desired semantics, is naturally the *inverse* of this semantics-to-presentations connection: We propose to tackle with large scale integration by developing some reverse analysis (Figure 3, bottom) which holistically analyzes the shallow clues, as guided by the hidden regularities, to discover the desired semantics.

This paper presents our view of shallow, holistic integration as a general approach for large-scale integration. To begin with, we motivate our approach by our survey of the deep Web, which revealed "concerted complexity" across sources. Further, we present two evidences from our recent works [15, 6], each of which demonstrates a materialization of the general approach: 1) *Syntactic parsing* [15], for understanding Web query interfaces (task T1), by the hypothesis of a hidden syntax. 2) *Statistical model discovery* [6], for matching interfaces across different sources (T2), by the hypothesis of a hidden generative model. In other words, as its main contribution, this paper aims at synthesizing the shallow integration approach as the generalized insight underlying these two different tasks and beyond.

The rest of this paper is organized as follows: Section 2 motivates the shallow techniques for large scale integration. Section 3 and Section 4 briefly present our two evidences respectively: query interface understanding and matching. Section 5 concludes the paper.

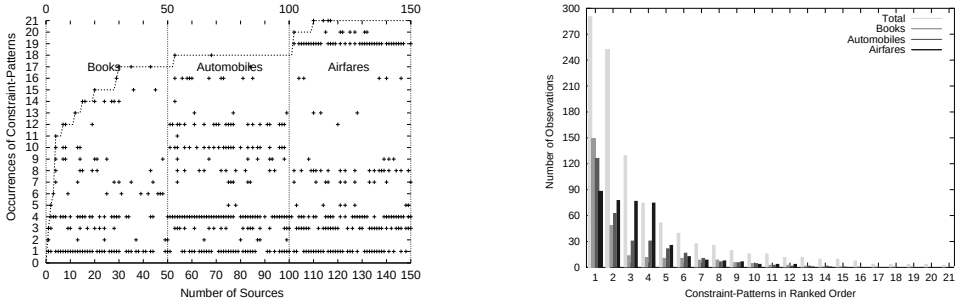
2 Shallow Integration

Our goal of the MetaQuerier, for integrating Web databases at a large scale, has largely remain unexplored. On one hand, for structured sources, *information integration* has mainly assumed relatively small-scaled, pre-configured systems (*e.g.*, Information Manifold [8], TSIMMIS [13]). On the other hand, research efforts on large-scale "metasearch" has focused on *text* sources (*e.g.*, source characterization [3] and query routing [11]). In contrast, we aim to enable *large-scale* metaquery over *structured*



(a) Growth of attribute vocabulary in each domain. **(b)** Frequencies over ranks for all the attributes.

Figure 4: Regularity of attribute vocabularies across sources.



(a) Growth of condition patterns over sources. **(b)** Frequencies over ranks for condition patterns.

Figure 5: Regularity of query conditions across sources.

databases.

With the virtually unlimited amount of information, while the deep Web is clearly an important frontier for data integration, this "wild" frontier of the deep Web is also relatively unexplored. To begin with, as observations, to better understand the challenges as well as the accompanying opportunities, we report an extensive "complexity" survey of this frontier. Further, as implications, we propose a new philosophy of "shallow integration" for integration at large.

2.1 Observations: Concerted Complexity

Our survey focuses on *query interfaces* (or query *forms*, as Figure 1 shows) of deep Web sources– Since data must be retrieved with queries, any attempts for integration must essentially interact with query interfaces.

Our survey studies sources in several representative domains: We manually collected about 400 sources using Web directories (*e.g.*, InvisibleWeb.com, BrightPlanet.com, WebFile.com) and search engines (*e.g.*, Google.com). The dataset[3] contains about 50 sources in each of the eight domains: Airfares, Automobiles, Books, Car Rentals, Hotels, Jobs, Movies, and Music Records.

Overall, our observations indicate remarkable "concerted complexity" among Web sources, in the same domain or across different ones: While sources proliferate, their aggregate "complexity" does not grow indefinitely, but instead demonstrates certain statistical regularities. In particular, we observe "converging" behaviors of the attribute vocabularies and query patterns across Web sources.

Converging attribute vocabularies: We first analyze *query attributes*, as the "vocabulary" (*e.g.*, author, title, make) for source query schemas . We observe that the aggregate vocabulary of attributes in the same domain tends to converge at relatively small size. Figure 4(a) analyzes the growth of vocabularies as sources increase in numbers. The curves clearly indicate the convergence of vocabularies. For instance, for Automobiles, 90% (65/72) attributes are observed at 57^{th} sources out of 101 sources. Since the vocabulary growth rates (*i.e.*, the slopes of these curves) decrease rapidly, as sources proliferate, their vocabularies will tend to stabilize. Note that the sources are sorted in the same order as they were collected without any bias.

In fact, the vocabularies will converge more rapidly, if we exclude "rare" attributes. To quantify, let the frequency of an attribute be the number of sources in which it occurs. Figure 4(b) orders these frequencies for all the attributes over their ranks. It is interesting but perhaps not surprising to observe that the distribution obeys the Zipf's law: The frequencies are inversely proportional to their ranks. Many low-ranked attributes thus rarely occur; in fact, 48% (203/422) attributes occur in only one source. Further, frequent attributes dominate: we observe that the top-20 attributes, or 4.7% (20/422) attributes, constitute 43% (1291/2992) of all the occurrences. What are the most "popular" attributes across all these sources? The top 5 frequent attributes are, in this order, title, keyword, price,

[3]We publish this dataset as the TEL-8 dataset in the UIUC Web Integration Repository [5].

make, and artist.

Converging condition patterns: We further analyze the "building blocks" for query interfaces– *i.e.*, the atomic conditions in query forms. For instance, as Figure 1 shows, *cars.com* has five conditions (on make, model, *etc.*) and *apartments.com* four (on city, state, *etc.*). Observe that these conditions seem to share some common "patterns": Those *condition patterns* present query conditions in certain visual arrangement (or layout)– Figure 7(c) shows several examples. For instance, pattern 1 represents a common format for conditions of the form [attribute; contains; text], by arranging attribute to the *left* of a textbox. Such conditions represent keyword search (by an implicit contains operator) on a textual attribute (*e.g.*, author).

Our survey finds that these condition patterns again reveal some concerted structure. There are only 25 condition patterns overall– which is surprisingly small as a vocabulary for online queries. As just mentioned, Figure 7(c) shows several frequently-used patterns. The distribution is again extremely non-uniform: Figure 5(b) ranks the patterns according to their frequencies (and omits 4 rare attributes in the tail, which occur only once in 150 sources), for each domain and overall. We observe again a characteristic Zipf-distribution, which confirms that a small set of top-ranked patterns will dominate.

We again observe the convergence behavior, both within and across domains. Figure 5(a) summarizes the occurrences of patterns. (To simplify, it similarly omits the rare "only-once" patterns.): The figure marks (x, y) with a "+" if pattern y occurs in source x. Like Figure 4(a), as more sources are seen (along the x-axis), the growth (along y) slows down and thus the curve flattens rapidly. Further, unlike Figure 4(a), we observe that the convergence generally spans across different domains, which indicates that most condition patterns are quite generic and not domain specific.

2.2 Implications: Shallow Integration Holistically

We have observed that, while sources proliferate, they demonstrate concerted complexity on various aspects. The observations motivate us to tackle the deep Web integration with a shallow but holistic paradigm. As our key insight, this new approach essentially leverages the challenge of large scale as an opportunity for holistic integration.

To begin with, our philosophy builds upon two hypotheses. As Section 1 discussed, any integration task (*e.g.*, T1) is essentially the discovery of certain target *semantics* (for T1: query capabilities of a source). While our goal is such underlying semantics, we can only observe some "surface" *presentations* (for T1: query forms presented in HTML). Our hypotheses conjecture how underlying semantics relates to observable presentations, across many sources.

($\mathcal{S}$) Shallow observable clues: The "underlying" semantics often connects to the "observable" presentations, or shallow clues, in some way of *connection*. That is, we can often identify certain observable clues (*e.g.*, occurrences of attributes, visual layout of query conditions), which reflect the underlying semantics.

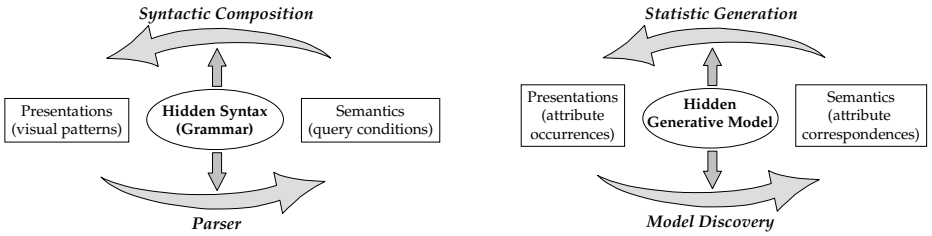
($\mathcal{H}$) Holistic hidden regularities: Such semantics-to-presentations connections often follow some implicit properties, or hidden regularities, which will reveal holistically across sources. That is, by observing many sources, we can often identify certain hidden regularities that guide how the underlying semantics connects to the presentations.

These hypotheses naturally explain our observations of the concerted complexity (Section 2.1). Since semantics connects to shallow clues (Hypothesis $\mathcal{S}$), in some regular ways across holistic sources (Hypothesis $\mathcal{H}$), those sources with similar semantics will naturally share some regularities at their presentations. In particular, consider the convergence phenomena of Section 2.1: First, we observed that sources in the same domain share a small vocabulary of attributes– *i.e.*, the *occurrences* (as shallow clues) of attributes across similar sources tend to follow some *repetitions* (as holistic regularities). Second, we observed that various query interfaces share a small set of query patterns– *i.e.*, the *visual arrangements* (as shallow clues) of similar query conditions tend to follow some *patterns* (as holistic regularities). (Section 3 and 4 will further explain these regularities, with more specific target semantics.)

These hypotheses shed new light on a different way for coping with information integration: As Figure 3 shows, our integration task, or the discovery of the desired semantics, is naturally the *inverse* of this semantics-to-presentations connection– We propose to tackle with large scale integration by developing some *reverse analysis* which holistically analyzes the shallow clues, as guided by the hidden regularities, to discover the desired semantics.

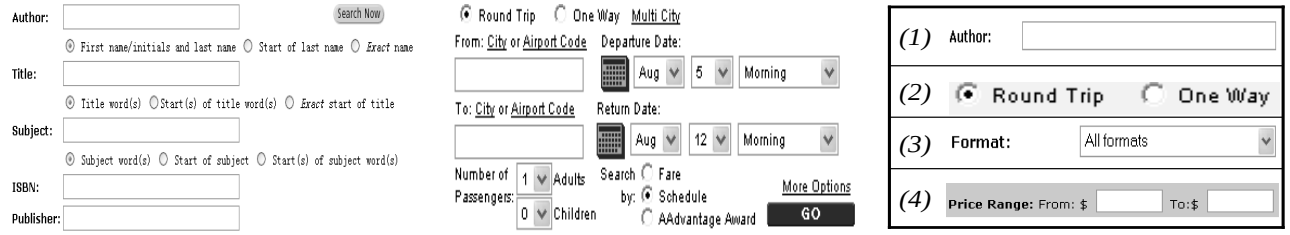
As evidences for materializing this shallow integration philosophy, we have studied two critical integration tasks [15, 6]: query interface understanding (T1 of Section 1) and matching (T2)– as Figure 6 illustrates and contrasts: On one hand, these evidences demonstrate how the general approach can be materialized for different integration tasks, where different types of target semantics are desired. On the other hand, these evidences contrast how the general approach can capture different types of semantics-to-presentations connections– of syntactic and statistical relationships respectively.

We believe this shallow integration paradigm promising for large scale integration by essentially leveraging the challenge of scale as an opportunity, with two main advantages: First, *scalability*: By integrating a large number of sources holistically, rather than individually or pairwise, we will be able to cope with the scale of integration, which is imperative in the new frontier of networked databases. Second, *solvability*: The large scale can itself be a crucial leverage to solve integration tasks. Our holistic approach will take advantage of the large scale (with sufficient "samples") for identifying the hidden regularities and applying principled holistic analysis.



(a) Evidence 1: query interface understanding. **(b)** Evidence 2: query interface matching.

Figure 6: Two materializations of the shallow integration approach.



(a) Query interface Q_{am}: *amazon.com*. **(b)** Query interface Q_{aa}: *aa.com*. **(c)** Examples of condition patterns.

Figure 7: Query interfaces examples.

3 Evidence 1: *Query Interface Understanding*

For integrating Web databases, the very first step is to "understand" what a query interface says (task T1, Section 1)– *i.e.*, what *query capabilities* a source supports through its interface, in terms of specifiable query conditions. For instance, *amazon.com* (Figure 7(a)) supports a set of five conditions (on author, title, . . ., publisher). Such query conditions establish the target *semantics* underlying a Web query interface that our task seeks to discover.

Such form understanding essentially requires both *grouping* elements hierarchically (*e.g.*, the query condition about author in *amazon.com* is a group of 8 elements: a *text* "author", a *textbox*, three *radio buttons* and their associated *text*'s) and *tagging* their semantic roles (*e.g.*, "author" has the role of an *attribute* and the textbox an *input domain*.) The tasks are challenging – it seems to be rather "heuristic" in nature with no clear criteria but only a few fuzzy *heuristics*, as well as *exceptions*. *First*, grouping is hard, because a constraint is generally n-ary, with various numbers of elements nested in different ways. ([*heuristics*]: Pair closest elements by spatial proximity. [*exception*]: Grouping is often not pairwise.) *Second*, tagging is also hard– There is no semantic labeling in HTML forms. ([*heuristics*]: A text element closest to a textbox field is its attribute. [*exception*]: Such an element can instead be an operator of this or next field.) *Finally*, with various form designs, their extraction can be inherently confusing.

Our solution is a specific materialization of the shallow integration approach (Figure 3): The observation of concerted condition patterns (Section 2.1) motivates us to hypothesize the existence of a *hidden syntax*, as the hidden regularities, across holistic sources. That is, we rationalize the concerted structure by asserting query-form creation as guided by some hypothetical syntax: As Figure 6(a) shows, the hypothetical syntax (as the hidden regularities) guides a syntactic composition process (as the connection) from query conditions (as the semantics) to their visual patterns (as the presentations). This hypothesis effectively transforms the problem into a new paradigm: We can view query interfaces as a *visual language* [10], whose composition conforms to a hidden, *i.e.*, *non-prescribed*, grammar. Their semantic understanding, as the reverse analysis, is thus a *parsing* problem.

This "language" paradigm enables principled solutions– to a problem that at first appears heuristic in nature– with the essential notions of a grammar and a parser:

- For *pattern specification*, the *grammar* provides a *declarative* mechanism. Such patterns (*e.g.*, Figure 7(c)) are simply declared by *productions* (*i.e.*, grammar rules) that encode their visual characteristics.

- For *pattern recognition*, the *parser* provides a *global* mechanism for systematically constructing a *parse tree* as a coherent interpretation of the *entire* query interface. Such a parse naturally groups elements (nested in subtrees) and tags their semantic roles (by grammar *symbols*), thus achieving form understanding.

For more details about the query interface understanding, please refer to [15].

4 Evidence 2: *Query Interface Matching*

Schema matching across Web interfaces (task T2, Section 1) is critical for mediating queries among deep Web sources. For instance, in Books domain, we may find subject is a synonym of category, and format is a synonym of binding. The target "semantics" to be discovered is thus the synonym correspondences of attributes.

Most schema matching works focus on finding *pairwise* attribute correspondences between two sources. For instance, traditional binary or n-ary [12] schema integration methodologies (as [1] surveys) exploit pairwise-attribute correspondence assertions (mostly manually given) for merging two or some n sources. Further, recent works on automatic schema matching mostly focus on matchings between two schemas (*e.g.*, [9]). The latest survey [14] thus abstracts schema matching as pairwise similarity mappings between two input sources. However, our scenario of deep Web integration calls for large scale matching, which cannot be adequately addressed by pairwise techniques.

Our solution is another materialization of the shallow integration approach (Figure 3), by holistically matching a set of sources. The observations of converging attribute vocabularies lead us to hypothesize the existence of a hidden generative model, as the hidden regularities, which probabilistically generates, from a finite vocabulary, the schemas we observed. Intuitively, such a model gives the statistical properties that constrains how synonym attributes may co-occur across interfaces. As Figure 6(b) shows, the hidden generative model (as the hidden regularities) guides a statistic generation process (as the connection) from attribute correspondences (as the semantics) among the vocabulary to their occurrences in interfaces (as the presentations). The reverse analysis is thus the discovery of such a hidden statistic model (which embeds attributes correspondence relationships), given a set of query schemas as statistic "observations."

To realize such hidden model discovery, we have proposed a general abstract framework, MGS, with three steps: (1) *Hypothesis modeling*: We first specify a parameterized structure of the hypothetical hidden models. Such models should capture the specific "synonym" semantics we want to discover. (2) *Hypothesis generation*: We then generate all "consistent" models that instantiate the observed schemas with non-zero probabilities. (3) *Hypothesis selection*: Finally, we select hypotheses that are consistent with the observed schemas with sufficient statistical significance. Please refer to [6] for details.

5 Conclusion

Toward building the MetaQuerier over the deep Web, where the integration is large scale, dynamic, and thus necessarily on-the-fly, this paper proposes our philosophy of shallow integration across holistic sources as a general approach. Motivated by the concerted-complexity observations of Web sources, our insights hinge on the hypotheses that the target semantics to be discovered often connects to certain shallow observable clues, in a way guided by some holistic hidden regularities across many sources. Integration is thus a reverse analysis, which holistically analyzes the shallow clues to discover the underlying semantics.

As concrete evidences, we have studied two different integration tasks [15, 6], each of which materializes shallow integration with syntactic or statistic approaches. Our experience indicates the promise of such techniques: For query interface understanding (Section 3), our experiment shows that the parsing approach achieves above 85% accuracy for extracting query conditions across randomly selected deep Web sources. For query interface matching (Section 4), our experiment in four popular domains (Books, Movies, Music Records and Automobiles) shows that the MGS framework can achieve over 90% accuracy.

Overall, we believe the shallow integration framework can effectively exploit large scale hidden regularities for achieving holistic integration. While we presented two evidences, such holistic approaches are well suited for the new frontier of large-scale networked databases in general and our focus of the deep Web in particular. As our key insight, in these settings, as sources proliferate, their aggregate complexity does not grow indefinitely– Instead, holistic hidden regularities often naturally emerge across sources. Exploiting such regularities, our shallow integration philosophy thus leverages the large scale challenge as an opportunity for new holistic techniques. We are eager to further apply this approach for building the MetaQuerier for the deep Web.

References

[1] C. Batini, M. Lenzerini, and S. B. Navathe. A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys*, 18(4):323–364, 1986.

[2] M. K. Bergman. The deep web: Surfacing hidden value. Technical report, BrightPlanet LLC, Dec. 2000.

[3] J. P. Callan, M. Connell, and A. Du. Automatic discovery of language models for text databases. In *SIGMOD Conference*, 1999.

[4] K. C.-C. Chang, B. He, C. Li, and Z. Zhang. Structured databases on the web: Observations and implications. Technical Report UIUCDCS-R-2003-2321, Department of Computer Science, UIUC, Feb. 2003.

[5] K. C.-C. Chang, B. He, C. Li, and Z. Zhang. The UIUC web integration repository. Computer Science Department, University of Illinois at Urbana-Champaign. http://metaquerier.cs.uiuc.edu/repository, 2003.

[6] B. He and K. C.-C. Chang. Statistical schema matching across web query interfaces. In *SIGMOD Conference*, 2003.

[7] B. He, Z. Zhang, and K. C.-C. Chang. Knocking the door to the deep web: Integrating web query interfaces. In *SIGMOD Conference, System Demonstration*, 2004.

[8] A. Y. Levy, A. Rajaraman, and J. J. Ordille. Querying heterogeneous information sources using source descriptions. In *VLDB Conference*, 1996.

[9] J. Madhavan, P. A. Bernstein, and E. Rahm. Generic schema matching with cupid. In *VLDB Conference*, 2001.

[10] K. Marriott. Constraint multiset grammars. In *Proceedings of IEEE Symposium on Visual Languages*, pages 118–125, 1994.

[11] W. Meng, K.-L. Liu, C. T. Yu, X. Wang, Y. Chang, and N. Rishe. Determining text databases to search in the internet. In *VLDB Conference*, 1998.

[12] S. Navathe and S. Gadgil. A methodology for view integration in logical data base design. In *VLDB*, 1982.

[13] Y. Papakonstantinou, H. García-Molina, and J. Ullman. Medmaker: A mediation system based on declarative specifications. In *ICDE Conference*, 1996.

[14] E. Rahm and P. A. Bernstein. A survey of approaches to automatic schema matching. *VLDB Journal*, 10(4):334–350, 2001.

[15] Z. Zhang, B. He, and K. C.-C. Chang. Understanding web query interfaces: Best effort parsing with hidden syntax. In *SIGMOD Conference*, 2004.