EntityAuthority: Semantically Enriched Graph-Based Authority Propagation

Julia Stoyanovich*
Columbia University
jds1@cs.columbia.edu

Srikanta Bedathur Klaus Berberich Gerhard Weikum
Max-Planck Institute for Informatics
{bedathur, kberberi, weikum}@mpi-inf.mpg.de

ABSTRACT

This paper pursues the recently emerging paradigm of searching for entities that are embedded in Web pages. We utilize information-extraction techniques to identify entity candidates in documents, map them onto entries in a richly structured ontology, and derive a generalized data graph that encompasses Web pages, entities, and ontological concepts and relationships. We exploit this combination of pages and entities for a novel kind of search-result ranking, coined EntityAuthority, in order to improve the quality of keyword queries that return either pages or entities. To this end, we utilize the mutual reinforcement between authoritative pages and important entities. This resembles the HITS method for Web-graph link analysis and recently proposed ObjectRank methods, but our approach operates on a much richer, typed graph structure with different kinds of nodes and also differs in the underlying mathematical definitions. Preliminary experiments with topic-specific slices of Wikipedia demonstrate the effectiveness of our approach on certain classes of queries.

1. INTRODUCTION

In this paper we present EntityAuthority: a framework for semantically enriched authority propagation on graphs that combine Web pages, entities, and ontological concepts, in order to improve the ranking of keyword query results. Current link analysis methods, such as PageRank and HITS, use the page-level Web graph for authority propagation [6, 29]. This approach is insufficient and, in our opinion, fundamentally inappropriate for the emerging style of "semantic" Web search which aims to provide users with entities (e.g., products or scholars) and semantic attributes rather than pages[2, 7, 8, 11, 13, 18, 25, 30, 31]. Recently, techniques have been proposed for analyzing link structures and computing rankings of objects in relational databases or in entity-relationship graphs [2, 3, 9, 11], but these methods do not consider Web data or do no longer connect the object ranking back to the Web pages where the objects appear. In contrast, we jointly consider both the page-level and entity-level linkage structures and their *mutual-reinforcement* cross-relationships, and utilize it for ranking the results of keyword queries against Web data. We focus on keyword search as this is still by far the most convenient way of information retrieval for most users. But we consider two kinds of result granularities: queries can return either pages or entities, based on the user's context and preference.

We utilize information-extraction techniques [1, 14, 16, 19] to identify semantic entities embedded in the text of Web pages, and to transform the page-level graph into a *generalized data graph*, with typed nodes that represent pages or entities, and with typed and weighted edges. We keep the entity part of this graph as noise-free as possible by mapping entities to nodes in a richly structured high-quality ontology with entities, concepts, and semantic relationships. More specifically, we use the popular GATE/ANNIE toolkit (`http://gate.ac.uk/`) for named-entity recognition and various heuristics for entity disambiguation. For the ontology part of our generalized data graph we employ the YAGO knowledge base which combines information from the Wikipedia category system and the WordNet thesaurus into a rigorously structured ontology with very high accuracy [34].

We assume and aim to exploit that there is mutual reinforcement between the authorities of pages and entities: pages become more valuable if they contain highly authoritative entities, and entities become more important if they are mentioned by highly authoritative pages. This resembles the HITS method [27] for Web-graph link analysis and the ObjectRank method [3], but our approach operates on a generalized data graph that gives us a much richer substrate for ranking.

Consider for example the query "NBA team". When issued against a leading search engine on March 31st, 2007, this query returned links to the official homepage of the National Basketball Association, to the NBA-related area of ESPN.com, and to several team directories. The results were highly relevant but not specific to the user's information need: it was not apparent from the top-10 which teams play in the NBA, and there were no links to any of the teams' homepages. We observe that super-authorities dominated the query result, and team homepages did not start appearing until rank 38. (Results were very similar for the re-phrased query "National Basketball Association team".) We consider this and similar queries in our experimental evaluation and demonstrate how our EntityAuthority ranking methods can benefit such situations.

The results of such queries can be either entities or pages. The former makes sense for a skilled user who has the proper context to interpret a concise list with entries such as "Miami Heat", "Dallas Mavericks", "Chicago Bulls", etc. The latter makes sense for a non-expert user who wants to see an entire textual page, with relevant entities highlighted and contextual information at one glance.

*This research was supported in part by NSF grant IIS-0121239 and by NIH grant 1U54CA121852-01A1.

Proceedings of the 10th International Workshop on Web and Databases (WebDB 2007), June 15, 2007, Beijing, China

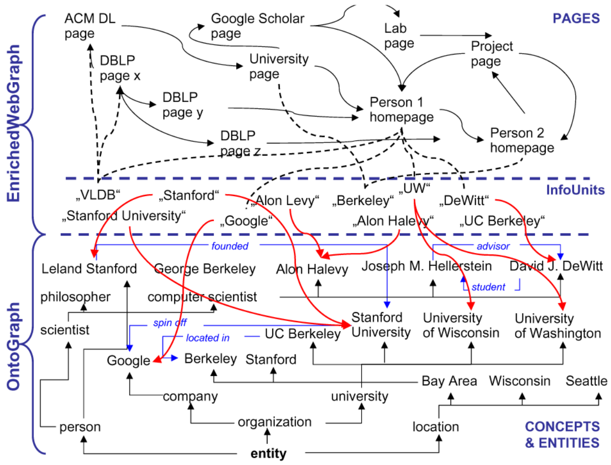

Figure 1: Running example: GeneralizedDataGraph (GDG)

EntityAuthority ranking is beneficial for this kind of Web-page ranking, too, by exploiting the semantic connections and mutual authority propagation among pages, entities, and concepts.

The EntityAuthority framework is fully implemented in an experimental prototype system. This paper makes the following novel contributions:

- We define the *GeneralizedDataGraph* that combines Web pages and entities and connects them to ontological concepts, as a rich substrate for query-result ranking.
- We define several new models and algorithms for authority propagation and derived rank measures, for both entity and page granularities.
- We present a prototype implementation, showing how to integrate information extraction with a rich ontology as the basis for semantically enriched query processing and ranking.
- We experimentally demonstrate the effectiveness of our approach on certain classes of queries against thematically focused samples of Wikipedia pages.

2. DATA MODEL

In this work, we operate on a directed graph data model that integrates standard Web pages and their link structure with entities extracted from these pages and semantic relationships between them. Figure 1 illustrates the key components in our data model using a toy example based on Web pages related to database research. As shown in the example, our model consists of three parts: EnrichedWebGraph (EWG) that is derived using the underlying Web data collection, OntoGraph – a richly structured ontological resource such as YAGO, and OntoMap – a semantic layer that connects EWG and OntoMap. We refer to an instance of the model consisting of EWG, OntoMap and the underlying ontological input as *GeneralizedDataGraph* (GDG) in the rest of this paper.

Formally, EWG is a directed, labeled, and weighted graph $G_D = (V_D, E_D, L_v)$ where V_D is a set of nodes, $E_D \subseteq V_D \times V_D$ is a set of edges between nodes, and L_v is a set of node annotations. Each node $v \in V_D$ is assigned a label $l_v \in L_v$. These labels can simply be Page, InfoUnit, or richer concept names such as Person, Organization, etc., that can be obtained through a sophisticated information extraction software. Each edge $e = (v_1, v_2)$ also carries a weight whose value depends on the type of nodes v_1 and v_2.

EWG is the outcome of enriching the standard page-level content and link structure with entities that can be extracted from each page individually. Thus each node in the EWG corresponds to either a Web page or an information unit (InfoUnit for short). An InfoUnit is a (short) textual snippet that is annotated as an instance of an entity by an information extraction tool such as GATE/ANNIE. This should be contrasted with the notion of an ontological entity/concept, which is the outcome of corpus-wide reconciliation of InfoUnits. To make this distinction clear, consider two pages P_1 and P_2 that mention “UW” and “University of Washington” respectively. During the initial stage of information extraction, these text snippets are identified as (potential) entities and are thus marked as InfoUnits. Note that no effort at this stage is made to determine whether these two snippets refer to the same entity or not.

We model the underlying ontological resource as a directed, weighted graph $G_O = (V_O, E_O, L_e)$ on a set of nodes V_O which correspond to entities/concepts and E_O is a multi-set of edges between these entities. (Note that we make no distinction between instances and concepts in this work.) Each edge in E_O is annotated with a label from the set of labels, L_e that indicates the relationship between entities connected by the edge. For example bornIn, locatedIn, instanceOf etc. are some of the labels available in YAGO – the precompiled ontology that we use. Each of these edges is assigned a weight representing the ontology's confidence in the relationship.

The next component in our model, OntoMap, forms the layer connecting the InfoUnits extracted from the corpus with entity/concept nodes in the ontology. OntoMap models a collection-wide entity reconciliation process that maps InfoUnits onto entities/concepts of the OntoGraph (illustrated in Figure 1 using solid edges from InfoUnits to OntoGraph layer). In order to support inherent ambiguities in the entity reconciliation process, our model allows for mapping of a single InfoUnit to more than one concept/entity. For example, consider the InfoUnit “UW” which is mapped to both “University of Wisconsin” or “University of Washington”.

3. ENTITY AUTHORITY

This section presents different models for computing page and entity authority values. We consider two major alternatives for the graph structure on which we compute such measures, and we discuss several models for performing these computations. The authority values of pages, entities, and concepts can be used for either statically ranking these units by query-independent importance or for ranking query results that are determined by an initial content-only retrieval procedure.

Structure of Generalized Data Graph.

The GeneralizedDataGraph (GDG) that we obtain from extracting entities from pages, mapping them to the ontology, and connecting entities and concepts by semantic relations provides us with a rich cross-reference structure that reflects the authority of both pages and entities. This graph consists of the following kinds of typed and weighted, directed edges, where all weights are normalized so that the weights of a node's outgoing edges sum up to one:

- hyperlinks between pages normalized by the source page's outdegree (as in standard PageRank),
- edges from a page to each entity or concept that has been extracted from the page, weighted by the confidence in the extraction and mapping to the corresponding ontology node,
- edges from entities to the concepts to which they belong, weighted with the ontology's confidence in the relationship and typed as an *isa* edge,

- edges from an entity to each of the pages where it appears (i.e., backward pointers of the information extraction and mapping process), weighted by the extraction and mapping confidence.

We do not currently include edges among entities and concepts, typed by their semantic relationship (e.g., *bornIn*, *locatedIn*, *childOf*, etc.). Including such edges is ongoing work. Note that the weights of the pointers from entities to pages are not necessarily the same as those from pages to entities because of the normalization step (a page may contain only a few entities, but these entities may appear in many pages). One immediate candidate for computing some notion of importance on the GDG would be to apply PageRank, or strictly speaking a weighted-edges variant of PageRank [5, 6] as if all nodes of the graph were pages. But we will see below that we can apply mathematically related but semantically more meaningful computations on such a richer graph.

Query Result Graph.

Alternatively, we can consider at query-time, a query-specific graph structure for computing, now query-dependent, authority measures. Suppose we first evaluate a keyword query against both pages and ontology nodes and compile a list of qualifying items. We consider an item as qualifying if it contains the query words or has a compact neighborhood in GDG that contains them (more details in Section 4.2). The resulting items - pages, entities, and concepts - can already carry initial *relevance scores* that reflect the frequencies or other kinds of prominence of keywords in the contents of the items or their neighborhoods. We now construct a *QueryResultGraph* (QRG for short) from these results, and the relevance scores can be considered in the edge weights of this graph.

The QRG consists of all initial results whose relevance score is above some predefined threshold (which may be set to 0, thus capturing all qualifying items). For each of these items we determine all their predecessors and all their successors in the GDG (these are nodes that point/are pointed to by the initial results), and add these nodes to the QRG. Finally, we look up all successors of predecessors and all predecessors of successors and add them, too. This set of GDG nodes constitutes the node set of the QRG, and the edge set of the QRG is simply the set of all GDG edges that connect two nodes that are both in the QRG. This query-dependent graph construction is similar to the procedure proposed in the original HITS method [27]. Edge weights can optionally be re-scaled by multiplying the weight of an edge $x \rightarrow y$ with the relevance score of the target node y, and some minimum value μ if y was not among the initial query results. This is similar to a biased random-walk model in which a hypothesized Web surfer prefers target nodes whose contents has high relevance scores. Regardless of whether this re-scaling is applied or not, edge weights need to be re-normalized because not all outgoing edges of a GDG node need to be present in the QRG.

Authority Measures.

The following authority measures can be computed on either one of the two above graphs. They utilize different extents of information captured by the graph constructions. The first measure uses only the page nodes of the graphs and then propagates page authority to entities and concepts; it is thus called *Page-Inherited Authority (PIA)*. The second measure, which is much more elaborate and one of this paper's main contributions, uses the full graph and is coined *Entity-deriVed Authority (EVA)*.

For PIA we consider only the pages in the GDG or the QRG and their incident page-to-page edges. We compute either (edge-weighted) PageRank or the page authority of the HITS method for each page. Let $\mathcal{A}_P(p)$ denote the authority of page p, and let $\mathbf{P}(x)$ denote the set of pages that point to ontology node x in the GDG (i.e., the pages from which x has been extracted). Further, let $w(x \rightarrow y)$ denote the *weight* of the edge from x to y. Then we define the authority of x as:

$$\mathcal{A}_P(x) = \sum_{p \in \mathbf{P}(x)} \mathcal{A}_P(p) \times w(p \rightarrow x)$$

So in the PIA model, an entity or concept is important if it appears in important pages, and page importance is predetermined by a page-links-only authority model. But we could also argue that a page is only important if it mentions important entities or concepts. This consideration leads to a mutual-reinforcement model between pages and entities. For example, a Web page such as `www.cs.stanford.edu` is important because it mentions high authorities like *Jennifer Widom* or *Hector Garcia-Molina*, and these are in turn high authorities because they are referenced by many important Web pages such as `Citeseer-top-authors`, `DBLP-top-authors`, `www.sigmod.acm.org` etc. This situation is captured by the following mutually recursive equations that define $\mathcal{A}_P$ – the page-level authority score, and $\mathcal{A}_O$ – the entity/concept-level authority score:

$$\mathcal{A}_P(p) = \sum_{x \in \mathbf{O}(p)} \mathcal{A}_O(x) \times w(x \rightarrow p) + \sum_{q \in \mathbf{P}(p)} \mathcal{A}_P(q) \times w(q \rightarrow p)$$

$$\mathcal{A}_O(x) = \sum_{p \in \mathbf{P}(x)} \mathcal{A}_P(p) \times w(p \rightarrow x) + \sum_{y \in \mathbf{O}(x)} \mathcal{A}_O(y) \times w(y \rightarrow x)$$

where p and q are pages, x and y are entities or concepts, $\mathbf{P}(z)$ denotes the set of pages to which a page, entity, or concept points, and $\mathbf{O}(z)$ denotes the set of ontology nodes to which a page, entity, or concept points.

This model is mathematically related to the HITS model that has mutual reinforcement between hubs and authorities, but the semantic interpretation of EVA is very different from HITS and the richer and heterogeneous graph structure that we use in EVA also makes the computation itself more demanding.

The authorities of pages and ontology nodes, $\mathcal{A}_P$ and $\mathcal{A}_O$ can be iteratively computed by the Orthogonal Iteration method [26]. In linear-algebra notation, the two equations for $\mathcal{A}_P$ and $\mathcal{A}_O$ can be rewritten as follows. Let $\vec{P}$ be a vector of the pages' $\mathcal{A}_P$ values and $\vec{X}$ be a vector of the ontology nodes' $\mathcal{A}_O$ values. Suppose there are m pages and n ontology nodes in the underlying graph. Now let PP be an $m \times m$ matrix, PX an $m \times n$ matrix, XP an $n \times m$ matrix, and XX and $n \times n$ matrix with the following entries:

$$\begin{aligned}
PP_{ij} &= w(i \rightarrow j) && \text{for pages } i, j \text{ with } i \neq j \\
&= 0 && \text{for } i = j \\
PX_{ij} &= w(i \rightarrow j) && \text{for page } i \text{ and ontology node } j \\
XP_{ij} &= w(i \rightarrow j) && \text{for ontology node } i \text{ and page } j \\
XX_{ij} &= w(i \rightarrow j) && \text{for ontology nodes } i, j \text{ with } i \neq j \\
&= 0 && \text{for } i = j
\end{aligned}$$

Then the equations for $\mathcal{A}_P$ and $\mathcal{A}_O$ given above can be phrased in vector form as:

$$\vec{P} = XP \times \vec{X} + PP \times \vec{P} \tag{1}$$

$$\vec{X} = PX \times \vec{P} + XX \times \vec{X} \tag{2}$$

The computation is initialized by choosing arbitrary start vectors for P and X that are not linearly dependent of the eigenvectors of the matrices in Equations 1 and 2. It is not difficult to satisfy this start condition; for example, it is satisfied by choosing uniform start values in all but degenerate situations. Then we evaluate each of the two equations by substituting the current values of the two vectors in their right-hand sides; the new values for the left-hand sides then become the values for the right-hand sides of the next iteration. These steps are iterated until the changes of the two vectors, as reflected either by the vectors' L1 norms or the relative ranking of their top-100 elements, drop below some threshold and become negligible. After each iteration step, the two vectors are re-scaled (i.e., multiplied by a normalization factor) so that their L1 norms become equal to one. The latter step ensures the convergence and unique result of the Orthogonal Iteration method.

The *PIA* model computes a global authority score of an entity based on global authority of the page(s) from which the entity was extracted. In contrast, *EVA* models mutual reinforcement between pages and entities on the QRG. The final family of ranking methods, Un-Typed Authority (*UTA*), serve as the middle-ground between the two methods. Like EVA, UTA is invoked at query time and operates on the QRG. But, unlike EVA, UTA simply runs an edge-weighted version of a standard ranking algorithm (e.g. PageRank or HITS) on this graph, ignoring node types.

4. SYSTEM IMPLEMENTATION

Our prototype implementation was built and evaluated on a part of the English-language Wikipedia – the free on-line encyclopedia that in its entirety contains over 1.6 million articles (we use a January 2006 snapshot of Wikipedia).

4.1 Building the Generalized Data Graph

We use several existing tools to build the GeneralizedDataGraph, and while information annotation and extraction is not a major contribution of this work, we gained some insight into performing these tasks on a fairly large scale. We used the GATE/ANNIE toolkit to identify entities of types *Location*, *Person* and *Organization* in the corpus, and were able to extract over 1.2 million annotations as InfoUnits. Entity annotation was time-consuming, forcing us to limit our experimental evaluation to a relatively small corpus at this stage. However, while extraction may be a bottleneck in an academic setting, this process is highly parallelizeable, and can be done on a large scale if enough hardware is available.

As the next step we group together InfoUnits that refer to the same real-life entity. For example, we would unify two occurrences of the same string (e.g. "Michael Jordan") into a single entity. We also attempt to identify InfoUnits that do not match literally, but are nonetheless likely to refer to the same real-life entity, e.g. "Michael Jordan" and "Mike Jordan". We rely on a combination of heuristics to identify groups of InfoUnits with high degree of string similarity. We consider strings that are at least 4 characters in length, and that match according to the SecondString[15] JaroWinkler-TFIDF metric, with a similarity of at least 0.95 out of 1.0. Additionally, we use the highly-accurate *means* relationship in YAGO [34] to identify pairs of synonymous strings. Using these simple and efficient heuristics we map 1.2 million InfoUnits to 240 thousand entities.

Finally, we use the same string-based heuristics to map discovered entities to nodes in YAGO. If an entity maps to one or several nodes in the ontology, we add the appropriate mapping edges to the GDG (these are the solid arrows from InfoUnits to ontology nodes in Figure 1). If no node in the OntoGraph was identified as a target for mapping, we add discovered entities to the OntoGraph, placing them directly under *Person*, *Organization* or *Location*.

During the initial stages of this project, we considered more sophisticated (and less restrictive) ways of grouping together similar InfoUnits and of mapping such groups to ontology nodes. We eventually found that these techniques introduced a considerable amount of noise, causing topic drift. We conclude that more sophisticated context-aware tools are required to reliably match strings that do not pass the 0.95 similarity threshold. Using such tools (once they become available and are efficient enough to operate on a large scale) would benefit our method.

4.2 Query Processing

Our system processes keyword queries, and returns both pages and entities as query results. In this work we focus primarily on entity extraction and ranking, not on query processing, and we use a simple query processing method in our prototype implementation.

We store the GeneralizedDataGraph in an Oracle 10g RDBMS, and use Oracle Text to identify relevant pages and entities at query time. Matches are identified using a tf-idf-based algorithm that incorporates stemming and word proximity information. Relevant entries receive a non-zero relevance score from Oracle; we normalize this score to the $(0, 1]$ range.

A page is considered relevant to a query if its body contains all words present in the query. The final score of a page is the product of its authority and relevance scores.

We identify entities that are relevant to a query using the OntoGraph built from YAGO [34]. An entity is considered relevant to a query if: (a) the entity matches the query by name; (b) the entity has strong string similarity with a YAGO node that matches the query; (c) *thematic neighborhood* of the entity matches the query. Thematic neighborhoods are formed by combining all parents of an entity or concept, i.e. by following *is-a* and *instance-of* ontology edges. We make no distinction between entities and concepts during query processing.

Consider for example the Serbian basketball player Vlade Divac. YAGO assigns Vlade to several categories, including *Serbian basketball players*, *LA Lakers players*, and *Olympic competitors for Yugoslavia*. Vlade's thematic neighborhood includes all these category names. For this reason the entity *Vlade Divac* will be returned as a match for queries like "Serbian LA Lakers players" and "Olympic basketball competitors". Relevant entities are ranked according to their authority scores; the relevance score of the query with respect to the thematic neighborhood is discarded.

The identified relevant pages and entities are added to the QueryResultGraph. The graph is then expanded by including predecessors and successors. The entire QRG is used for ranking, but only the relevant pages and entities are returned as the query result.

5. PRELIMINARY EVALUATION

5.1 Experimental Setup

We evaluate the performance of our system on a subset of the English-language Wikipedia; we focus on two thematic slices: *Serbia* and *basketball*. Pages were included into the respective slice based on whether they contain the words "Serbia" and "basketball". The slices are comparable in size, and together include about 7800 Wikipedia articles.

We selected 20 keyword queries, 10 queries per slice, for our preliminary experiments. Queries ranged between 1 and 6 words in length. Some examples of queries are *lake*, *politician*, *physics*, *living writer prize winner* on the *Serbia* slice, and *NBA venue*, *college basketball*, *African American basketball player Olympic competitor* on the *basketball* slice. Queries were selected so as to have non-trivial recall in the ontology with respect to the slice: query terms had to match thematic neighborhoods of at least a few ontology nodes relevant to the slice. We allowed for significant variation in ontology recall: for some queries, hundreds of ontology nodes matched, while for others there were only a handful of matches. In

Method	DCG	NDCG	Recall	Precision
PIA	12.73	0.91	0.88	0.97
UTA-PR	12.67	0.91	0.88	0.96
UTA-HITS	11.17	0.78	0.82	0.91
EVA	11.95	0.86	0.90	0.99

Figure 2: Ranking on Entities

Method	DCG	NDCG	Recall	Precision
PR	4.14	0.34	0.42	0.59
UTA-PR	2.90	0.26	0.35	0.53
UTA-HITS	2.62	0.24	0.30	0.47
EVA	7.61	0.60	0.67	0.89

Figure 3: Ranking on Pages

the most extreme case, only one ontology node matched the query.

We compare four ranking methods in our experiments: one query-independent and three that re-rank results at query time. These are query-independent PageRank (PR) for pages, and the corresponding Page-Inherited Authority (PIA) for entities. Query-time reranking methods include Un-Typed Authority with PageRank (UTA-PR) and with HITS (UTA-HITS), and Entity-Derived Authority (EVA). We consider top-20 pages and top-20 entities returned by each ranking method in our evaluation.

For a given query, top-ranking results were collected and pooled. The quality of each result was then evaluated by two of the co-authors of this paper. Evaluators had no knowledge either of the method that retrieved the result, or of the result's rank. We used a simple *goodness* metric, a value between 0 and 2, in our evaluation, and averaged the goodness scores if there was disagreement. Goodness scores were assigned as follows: 0 for irrelevant, 1 for somewhat relevant or relevant but not very important, and 2 – for very relevant entries. To judge goodness of an entity, the evaluator was asked to identify a Wikipedia page that was most relevant to the entity, and use the contents of the page to guide the evaluation. We also considered using a combination of metrics, e.g. coverage and specificity, to evaluate the results. However, these metrics were designed with documents (or parts of documents) in mind, and it was not clear how to apply them to entities and concepts.

We use four metrics to assess the performance of the ranking methods: discounted cumulated gain (*DCG*) [24], normalized discounted cumulated gain (*NDCG*) [24], precision, and recall. All metrics were applied at top-20. DCG is a cumulative metric that weighs goodness scores by rank, penalizing entries that appear at later ranks. NDCG is an average metric that normalizes each entry in the DCG vector by the corresponding value in the ideal vector (see below). We report DCG in addition to NDCG because it incorporates recall (a low-recall method may have a perfect NDCG score, but it will not have a perfect DCG score). As suggested in [24], we use $log_2(rank)$ as the discount factor. For recall, we consider an entry to be relevant to a query if at least one of the two evaluators considered the entry relevant (goodness score $\geq$ 0.5). Precision is calculated with respect to the ideal: we pool together the relevant entries retrieved by all methods, order them by descending goodness scores, and calculate recall at top-20. Precision of a ranking method is then calculated as $recall_{method}/recall_{ideal}$.

5.2 Results and Discussion

Results of our experimental evaluation are summarized in figures 2 and 3. All methods that involve query-time reranking operate on the QueryResultGraph: a query-dependent subset of the GeneralizedDataGraph. Each entry represents an average among 20 queries in our experiments. We view results obtained by query-independent PageRank (*PR* entry in figure 3) as the base-line for our experiments. Please note that NDCG, precision, and recall are normalized, and that perfect DCG is 15.63 in our setting.

We make the following observations from these results.

- For the chosen queries, the set of highly-ranked entities consistently and significantly outperforms highly-ranked pages, according to all used metrics.
- EVA significantly outperforms other methods according to all metrics with respect to highly-ranked pages.
- No conclusion can be drawn about the relative performance of ranking methods with respect to entities. All methods produce high-quality results.

Wikipedia contains a significant number of hubs (list pages that link to pages relevant to a topic) and super-authorities (e.g. country and major city pages). Such pages attain high query-independent PageRank scores, and appear in very high ranks in response to most queries. So, for the query *basketball* on the *Serbia* slice, query-independent PageRank returns the following pages in the top-20: “1977”, “1990s”, “Greece”, and “Belgrade”. Re-ranking on the combined page-entity graph with UTA leads to page matches that still have high global authority, but are more focused on the query: “List of athletes by nickname”, “August 2004 in sports”, and even a high-quality match “Basketball at the 2004 Summer Olympics (team squads)”. Re-ranking with EVA returns pages that are very specific both to the query and to the slice: “Basketball in Yugoslavia”, “Vlade Divac”, “Basketball World Championship” and “National pastime”. Entity matches are even better, and include “Michael Jordan”, “LA Lakers”, “NBA”, “Madison Square Garden”, “Belgrade Arena”, and “Predrag Danilovic”. The name “Vlade Divac” was not recognized by GATE, and we miss out on an entity that corresponds to this Serbian national hero in the top-20.

Our novel ranking technique, EVA, is not realizing its full potential when ranking on entities, most importantly because our current extraction and mapping techniques do not allow us to include inter-entity edges into the QRG, limiting the flow of authority, particularly for entities. We plan to address this issue in future work.

The choice of relatively small thematic slices for our preliminary experiments makes the size of the GeneralizedWebGraph more manageable, but confines us to working with only a small subset of the ontology, limiting the recall of entity-based methods in some cases. For example, the query "orthodox monastery" on the Serbia slice was able to identify a single entity as a match – the monastery Hilandar. This happened because the ontology contained a limited amount of information relevant both to the query and to the thematic slice. The situation was further aggravated by the fact that the GATE/ANNIE Toolkit was not very successful when mining foreign-language names, and so many InfoUnits that could have been matched to relevant OntoGraph nodes went unnoticed.

We plan to enhance the recall of entity-based techniques by scaling up the experimental testbed, so as to be able to use a larger part of the ontology for query processing. However, in a highly dynamic environment such as the World Wide Web, content of pages constantly evolves, with new pages (and new entities/concepts) continually entering the picture. Building quality ontologies is non-trivial, and is bound to be reactive, i.e. happen *after* and in response to content evolution. For this reason, it is particularly important to not rely solely on entity-based methods, and to return page matches if few high-quality entities match the query.

6. RELATED WORK

Link analysis has come a long way since the seminal articles by Kleinberg [27] and Page et al. [33] were published. These early approaches and their extensions, overviews of which are given in [6, 28], are based on a simple directed graph model. More sophisticated (e.g., weighted, multi-type, and labeled) graph models were considered in [4, 12, 20, 22, 35]. More recently, the paradigm of link analysis has been carried over to graphs other than the Web graph, namely, relational databases with records and foreign-key relationships constituting the nodes and edges of the graph, and entity-relationship graphs that capture, for example, bibliographic data such as DBLP or Citeseer. Such settings have led to new forms of ObjectRank, PopRank, or EntityRank measures [2, 3, 9, 11, 32]. ObjectRank [3] resembles our approach because it is also inspired by HITS, but it does not address Web data at all. EntityRank [11] addresses the ranking of entities extracted from Web pages, but its focus is on frequency-based content strength and it does not consider the graph structure of the Web and its embedded entities. PopRank [32] is closest to our framework; it uses a "random object finder" model on an object-relationship graph and combines this with a prior popularity derived from pages' PageRank values (the latter is similar to our PIA method). Our approach is more powerful because we treat both pages and entities as first-class citizens in ranking, and because we also consider ontological relationships and confidence values from extraction and disambiguation.

Searching the Web at the finer and and semantically more expressive granularity of entities (Web objects) and their relationships, instead of the coarser page granularity prevalent today, has been pursued in different variants like faceted search, vertical search, object search, and entity-relationship search (e.g., [8, 10, 18, 25, 30, 32]. The approaches most closely related to our work are the Libra system [31, 32], the EntitySearch engine [11], and the ExDB system [7]. All three systems extract entities and relations from Web data and provide ranked retrieval. Libra uses a variety of techniques for ranking, including the PopRank model mentioned above and a record-level statistical language model; the EntitySearch engine mostly relies on occurrence-frequency statistics; ExDB factors extraction confidence values into its ranking but does not consider any link information. None of these ranking models considers the mutual authority propagation between entities and pages that we exploit in our Generalized Data Graph.

7. CONCLUSIONS AND FUTURE WORK

In this paper we developed an entity-aware ranking framework, and presented a novel ranking algorithm, EntityAuthority, that models the mutual reinforcement between pages and entities. We demonstrated how an ontology can be used for query processing in this setting. We presented a prototype implementation of our system and some preliminary experimental results that highlight the improvement in query result quality achieved by EntityAuthority, for both pages and entities.

In the future we plan to extend our work in several directions. We will work on adding inter-entity edges to the GeneralizedDataGraph in order to further enhance ranking on entities. We also plan to extend our experimental testbed and conduct an extensive experimental evaluation of our methods, focusing both on query result quality and on scalability of the framework.

8. REFERENCES

[1] E. Agichtein, S. Sarawagi. Scalable Information Extraction and Integration. Tutorial Slides, *KDD* 2006.

[2] K. Anyanwu, A. Maduko, A. P. Sheth. Semrank: Ranking Complex Relationship Search Results on the Semantic Web. *WWW* 2005.

[3] A. Balmin, V. Hristidis, Y. Papakonstantinou. ObjectRank: Authority-Based Keyword Search in Databases. *VLDB* 2004.

[4] G. Bhalotia et al. Keyword Searching and Browsing in Databases using BANKS. *ICDE* 2002.

[5] K. Bharat, M. R. Henzinger. Improved Algorithms for Topic Distillation in a Hyperlinked Environment. *SIGIR* 1998.

[6] A. Borodin et al. Link Analysis Ranking: Algorithms, Theory, and Experiments. *ACM TOIT.* 2005.

[7] M. J. Cafarella et al. Structured Querying of Web Text Data: A Technical Challenge. *CIDR* 2007.

[8] S. Chakrabarti. Breaking Through the Syntax Barrier: Searching with Entities and Relations. *ECML* 2004.

[9] S. Chakrabarti. Dynamic Personalized Pagerank in Entity-Relation Graphs. *WWW* 2007.

[10] K. C.-C. Chang. Large-Scale Deep Web Integration: Exploring and Querying Structured Data on the Deep Web. Tutorial Slides, *SIGMOD* 2006.

[11] T. Cheng, K. C.-C. Chang. Entity Search Engine: Towards Agile Best-Effort Information Integration over the Web. *CIDR* 2007.

[12] K. P. Chitrapura, S. R. Kashyap. Node Ranking in Labeled Directed Graphs. *CIKM* 2004.

[13] J. Chu-Carroll et al. Semantic Search via XML Fragments: a High-Precision Approach to IR. *SIGIR* 2006

[14] W. Cohen. Information Extraction. `http://www.cs.cmu.edu/~wcohen/ie-survey.ppt`

[15] W. Cohen, P. Ravikumar, S. Fienberg. A Comparison of String Distance Metrics for Name-Matching Tasks. *IIWeb* 2003.

[16] H. Cunningham. An Introduction to Information Extraction. *Encyclopedia of Language and Linguistics*, Elsevier, 2005.

[17] P. DeRose et al. Dblife: A Community Information Management Platform for the Database Research Community (Demo). *CIDR* 2007.

[18] A. Doan et al. Community Information Management. *IEEE Data Eng. Bull.* 2006.

[19] A. Doan, R. Ramakrishnan, S. Vaithyanathan. Managing Information Extraction: State of the Art and Research Directions Tutorial Slides, *SIGMOD* 2006.

[20] F. Geerts, H. Mannila, E. Terzi. Relational Link-based Ranking. *VLDB* 2004.

[21] N. Gövert et al. Evaluating the Effectiveness of Content-oriented XML Retrieval Methods. *Inf. Retr.* 2006.

[22] L. Guo et al. XRANK: Ranked Keyword Search over XML Documents. *SIGMOD* 2003.

[23] H. Hwang, V. Hristidis, Y. Papakonstantinou. ObjectRank: A System for Authority-based Search on Databases. *SIGMOD* 2006.

[24] K. Järvelin, J. Kekäläinen. Cumulated Gain-based Evaluation of IR Techniques. *ACM TOIS* 2002.

[25] M. Hearst Design Recommendations for Hierarchical Faceted Search Interfaces. *SIGIR Workshop on Faceted Search* 2006.

[26] D. Kempe, F. McSherry. A Decentralized Algorithm for Spectral Analysis. *STOC* 2004.

[27] J. M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM* 1999.

[28] A. N. Langville, C. Meyer. Deeper Inside PageRank. *Internet Mathematics* 2004.

[29] A. N. Langville, C. Meyer. Google's PageRank and Beyond: The Science of Search Engine Rankings. *Princeton University Press* 2006

[30] J. Madhavan et al. Web-Scale Data Integration: You can Afford to Pay as You Go. *CIDR* 2007.

[31] Z. Nie et al. Web Object Retrieval. *WWW* 2007.

[32] Z. Nie, et al. Object-level Ranking: Bringing Order to Web Objects. *WWW* 2005.

[33] L. Page et al. The PageRank Citation Ranking: Bringing Order to the Web. *Stanford University Tech. Report* 1998.

[34] F. M. Suchanek, G. Kasneci, G. Weikum. YAGO: A Core of Semantic Knowledge - Unifying WordNet and Wikipedia. *WWW* 2007.

[35] W. Xi et al. Link Fusion: A Unified Link Analysis Framework for Multi-type Interrelated Data Objects. *WWW* 2004.