

# TripleRank: Ranking Semantic Web Data By Tensor Decomposition

Thomas Franz, Antje Schultz, Sergej Sizov, and Steffen Staab

ISWeb - University of Koblenz-Landau, Germany,  
WWW home page: <http://isweb.uni-koblenz.de>  
{franz, antjeschultz, sizov, staab}@uni-koblenz.de

**Abstract.** The Semantic Web fosters novel applications targeting a more efficient and satisfying exploitation of the data available on the web, e.g. faceted browsing of linked open data. Large amounts and high diversity of knowledge in the Semantic Web pose the challenging question of appropriate relevance ranking for producing fine-grained and rich descriptions of the available data, e.g. to guide the user along most promising knowledge aspects. Existing methods for graph-based authority ranking lack support for fine-grained latent coherence between resources and predicates (i.e. support for link semantics in the linked data model). In this paper, we present TripleRank, a novel approach for *faceted* authority ranking in the context of RDF knowledge bases. TripleRank captures the additional latent semantics of Semantic Web data by means of statistical methods in order to produce richer descriptions of the available data. We model the Semantic Web by a 3-dimensional tensor that enables the seamless representation of arbitrary semantic links. For the analysis of that model, we apply the PARAFAC decomposition, which can be seen as a multi-modal counterpart to Web authority ranking with HITS. The result are groupings of resources and predicates that characterize their authority and navigational (hub) properties with respect to identified topics. We have applied TripleRank to multiple data sets from the linked open data community and gathered encouraging feedback in a user evaluation where TripleRank results have been exploited in a faceted browsing scenario.

## 1 Introduction

Relevance ranking is a crucial component for a wide range of Semantic Web applications, such as semantic search and semantic browsing. Online services such as the [dbpedia.org](http://dbpedia.org) [4] provide rich descriptions of Semantic Web resources that render manual browsing extremely difficult. For instance, the resource *The\_Beatles* has no less than 1228 links going in and out of it. In the context of RDF knowledge bases, data can be seen as a graph where nodes represent RDF resources and edges correspond to RDF predicates that link resources. Consequently, graph-based authority ranking algorithms known from Web retrieval, such as PageRank [8], HITS [23] or SALSA [27], can be adopted for the Semantic Web setting, too. Instead of ratings for Web pages they will then output ratings for RDF resources, with respect to one or more criteria, e.g. hub and authority scores in HITS. These scores reflect the centrality/importance of particular RDF resources in the knowledge representation and thus can be exploited for relevance estimation.

Two important observations can be made about the authority ranking for RDF data. On one hand, the computational models of standard algorithms for Web analysis only consider structural information, i.e. the connectivity of graph nodes. Additional link semantics, e.g. knowledge about different link types, is not used. On the other hand, knowledge representation in the Semantic Web is heterogeneous. There is no single ontology for the Semantic Web that describes all of the available data in a concise way. Instead, there are many cases of overlapping, redundant, and conflicting ontologies describing similar information. Therefore, we may expect redundancies like the co-existence of different predicates with highly similar (or identical) meaning. For instance, even the well maintained data from [dbpedia.org](http://dbpedia.org) contains the redundant links <http://dbpedia.org/ontology/writer> and <http://dbpedia.org/property/writer> which have the same semantics. Common authority ranking algorithms provide no support for finding such groups of semantically coherent relationships.

For incorporating missing link semantics, various adaptations to the common authority ranking models can be constructed. However, changes to the computational model may alter the original behavior of an algorithm and impose new restrictions. For instance, the ObjectRank [6] approach adds link semantics to PageRank. However, it requires to assign static probabilities to each type of link before its application. Thus, ObjectRank lacks the flexibility of the original approach which can be easily applied to arbitrary Semantic Web graphs without such upfront assignments. Moreover, the modification of the original models and algorithms can impose side effects and hampers the comprehensibility of the approach. Interpretations on the impact of each modification are required to enable the evaluation of such modifications.

In this paper, we introduce a new approach TripleRank for authority ranking of linked data on the Semantic Web that naturally takes into account its additional semantics. We reconsider the paradigm of two-dimensional graph representation (e.g. adjacency matrix for HITS, probabilistic transition matrix for PageRank, a mixture of these models for SALSA, etc.) and represent semantic graphs by 3-dimensional tensors. Our model has a clear semantics and supports the seamless representation of RDF graphs, including link/predicate semantics. For finding authoritative sources, we apply the PARAFAC decomposition [18] which can be seen as a multi-dimensional correspondent to a singular value decomposition of a matrix. Tensor decomposition yields rich information on the resources and predicates of the analyzed Semantic Web graph beyond simple rankings. Rated groupings of RDF resources and RDF predicates with respect to their (latent) topic, authority, and navigational (hub) characteristics are the outcome of the analysis. These results can be exploited to support a variety of applications, such as semantic faceted navigation. We present encouraging results of TripleRank gathered by its application to multiple RDF data sets from the Wikipedia free encyclopedia and the DBLP research publications database.

The contribution is organized as follows. In Section 2 we distinguish our work from related approaches for rating Web pages and (semi-)structured data. Section 3 introduces the tensor-based semantic graph representation for RDF knowledge bases and its PARAFAC decomposition for authority ranking. Section 4 addresses design and architecture issues of TripleRank. Consequently, Section 5 shows results of systematic user

studies that demonstrate the viability of TripleRank and its advantages in comparison with other methods. Section 6 summarizes and concludes the contribution.

## 2 Related Work

From the conceptual perspective, two topics can be seen as closely related to our TripleRank approach: authority ranking for Web contents and graph-based relevance ranking for semi-structured data. This section gives a short overview of these areas and distinguishes TripleRank from other existing solutions.

### 2.1 Rating Web Pages

PageRank [8], HITS [23] and SALSA [27] are prominent algorithms for ranking Web pages based on link analysis. PageRank builds upon a model of a random walk among Web pages, where the stationary probability of passing through a certain page is interpreted as measure of its importance. HITS is based on the notion of a mutual reinforcement between importance (authority) and connectivity (hub) scores of Web pages. SALSA can be seen as a more complex hybrid solution that integrates ideas of PageRank and HITS by combination of both link traversing directions (i.e. forward and backward) for constructing graph models. The conceptual generalization for this kind of methods is given in [14]. Unlike TripleRank, this family of methods provides no natural mechanisms for expressing and exploiting link semantics.

The contextualization of graph models can be achieved through different customizations of the mentioned models. Possible adaptations include various custom weightings of graph edges (e.g. based on appearance of particular terms in Web documents [31,29], content classification [13,19], structural properties like in-domain vs. out-domain linking [7], etc.) or joint probabilistic modeling for content and connectivity of Web pages [12]. In contrast to TripleRank, these solutions are designed for the Web setting and do not introduce distinguished link semantics. The solution presented in [26] uses for Web authority ranking the higher-order representation of the hyperlink graph by labeling the graph edges with the anchor text of the hyperlinks. This method is closely related to TripleRank, but addresses a fully different problem setting (links and anchors in the Web graph vs. linked data in Semantic Web RDF graphs). Additionally, our paper augments the introduction of TripleRank with a user evaluation that gives insight into the applicability of the overall authority ranking approach.

Another kind of contextualization for authority ranking models can be observed in the area of search personalization. For instance, Eirinaki and Vazirgiannis present a modification of the PageRank algorithm to compute personalized recommendations of Web pages given a path of visited pages [16]. Their approach requires access to web server logs that provide statistics about the paths browsed by other users. BrowseRank [28] is a further example of a page ranking approach that requires to collect statistics on user behavior such as the time spent on a web page. The generalized algorithm for personalized authority ranking is described in [22].

Our TripleRank approach is designed for a different scenario of resource recommendation when browsing linked data. As when browsing the Web, detailed statistical

information about prior user visits is not available in our problem setting. Our TripleRank approach is conceptually more general and does not rely on user profiles and query logs. As when browsing the Web, detailed statistical information about prior user visits is often not available in our Semantic Web scenario. However, this information can be easily integrated with TripleRank, if necessary.

## 2.2 Rating (Semi-)Structured Data

ObjectRank [6] adds authority transfer weights for different types of links to the PageRank algorithm. Such weights influence the *random walk* of prospective users and are to be assigned by domain experts. Beagle++ [10] is an extension for the Beagle desktop search engine that applies ObjectRank to RDF meta data about desktop objects to improve their ranking in desktop search scenarios. TripleRank also considers the semantics of link types, however, it is an approach for computing ranks for RDF resources at runtime, does not rely on manually assigned link weights, and is based on the generalized HITS algorithm instead of PageRank.

Several research works have dealt with information retrieval in the Semantic Web, presenting approaches for ranking resources based on keyword input, e.g. Swoogle [15] and ReConRank [21]. Browsing-based search strategies, as targeted by TripleRank, differ from keyword search with respect to the search process, its perception by users, and the input and output data processed to implement browsing support [9]. Rocha et al. [32] presented an approach to semantic search that associates tf-idf, represented by semantic specificity and semantic similarity, to semantic relations to apply a spreading activation approach on this augmented graph.

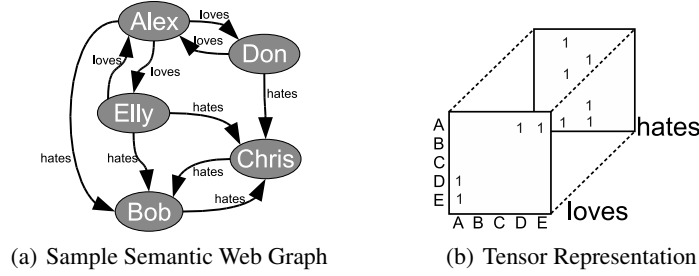
Anyanwu and Sheth present a framework for query answering with respect to so called semantic associations [3]. A semantic association represents semantic similarity between paths connecting different resources in an RDF model. Aleman-Meza et al. [1] presented and evaluated methods for ranking semantic associations. As a continued work of [3], the presented methods target the identification of similar resources to apply it in scenarios like terror-prevention. Their approach involves ranking criteria considering graph structure, and user context. User context is defined statically by selecting ontology concepts that are considered as representative for a user's context. Ramakrishnan et al. present heuristics for weighting graph patterns connecting two nodes in a graph considering the differences of edges given by RDF graphs that include schema information encoded as RDFS ontologies [30]. Prior approaches on graph pattern analysis presented methods assuming that only one type of edge exists. Next to a presentation of the heuristics, they present an evaluation of them targeting the question which heuristic results in higher quality patterns.

## 3 TripleRank: The Semantic Web as Tensor

The authority ranking approach by TripleRank builds upon the analysis of a tensor. In this section, we first introduce our modeling approach for representing the Semantic Web as tensor. We then continue with details on its analysis.

### 3.1 The TripleRank Model

We define a Semantic Web graph as a graph  $G = (V, L, E, linkType)$  where  $V$  is the set of RDF resources in the graph,  $L$  is the set of literals contained, and  $E$  is the set of links between RDF resources in  $V$ . Additionally, the function  $linkType : E \rightarrow V$  returns the URI of the property that links two resources. Fig. 1(a) shows a small Semantic Web graph that contains seven resources (A,B,C,D,E, loves, hates) and ten links of two different types: *loves* and *hates*.



**Fig. 1.** Modeling Example

We represent Semantic Web graphs by a 3-dimensional tensor  $\mathbf{T}$  where each of its slices represents an adjacency matrix for one RDF property. Figure 1(b) illustrates the tensor resulting from the transformation of the sample graph shown in Fig. 1(a). The first adjacency matrix  $\mathbf{T}(:, :, 1)$ <sup>1</sup> models linkage by the property *loves*. An entry  $> 0$  corresponds to the existence of a link by this property, empty entries are considered as zeroes. The second matrix  $\mathbf{T}(:, :, 2)$  models links by the property *hates*. For instance, the graph expresses that Alex *hates* Bob, which corresponds to  $\mathbf{T}(1, 2, 2) = 1$ .

### 3.2 PARAFAC for Authority Ranking

The Semantic Web graph can be described by an adjacency matrix. For a network graph matrix  $M$  the well known authority ranking methods like HITS [23] can be applied. HITS defines the authority ranking problem through mutual reinforcement between so-called hub and authority scores of graph nodes. The authority (relevance) score of each node is defined as the sum of hub scores of its predecessors. Analogously, the hub (connectivity) score of each node is defined as a sum of the authority scores of its successors. By applying the Singular Value Decomposition (SVD) to the adjacency matrix, we obtain hub and authority scores of graph nodes for each singular value of  $M$ , which can be interpreted as rankings regarding different themes or latent topics of interest. Formally, by this method, some arbitrary matrix  $M \in \mathbf{R}^{k \times l}$  is splitted into three matrices  $U \in \mathbf{R}^{k \times m}$ ,  $S \in \mathbf{R}^{m \times m}$ ,  $V \in \mathbf{R}^{l \times m}$ .  $U$  and  $V$  represent the outlinks and inlinks with respect to the principal factor contained in  $S$ . Corresponding to our

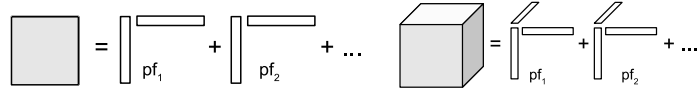
<sup>1</sup> Throughout this paper we use the common Matlab-notation for addressing entries in tensors and vectors

notation,  $M$  can be written as sum of rank-one-matrices by  $M = \sum_{k=1}^m S^k \cdot U^k \circ V^k$ . This 2-way decomposition yields authority and hub scores (cf. Fig. 2(a)) [25].

Modeling several link types by separate matrices results in very sparse and not connected matrices. Instead, the tensor model applied by TripleRank enables the representation of all adjacency matrices including information about the connections between link types. Tensor decomposition methods like PARAFAC can then detect further hidden dependencies.

These methods are regarded as higher-order equivalents to matrix decompositions. The PARAFAC tensor decomposition has the advantage of robustness and computational efficiency. These advantages are due to its uniqueness up to scaling and permutation of the produced component matrices [18]. By PARAFAC input tensors are transformed to so called Kruskal tensors, a sum of rank-one-tensors. Consequently, in TripleRank we derive authority and hub scores for particular latent aspects (topics) of the analyzed data from particular rank-one-tensors of the decomposition. In the context of this paper we focus on three-mode-tensors that represent connectivity between graph nodes together with semantics of links (predicates).

Formally, a tensor  $\mathbf{T} \in \mathbf{R}^{k \times l \times m}$  is decomposed by n-Rank-PARAFAC into components matrices  $U_1 \in \mathbf{R}^{k \times n}$ ,  $U_2 \in \mathbf{R}^{l \times n}$ ,  $U_3 \in \mathbf{R}^{m \times n}$  and  $n$  principal factors ( $pf$ )  $\lambda_i$  in descending order. Via these  $\mathbf{T}$  can be written as a Kruskal tensor by  $\mathbf{T} \approx \sum_{k=1}^n \lambda_k \cdot U_1^k \circ U_2^k \circ U_3^k$  where  $\lambda_k$  denotes the  $k$ th principal factor,  $U_i^k$  the  $k$ th column of  $U_i$  and  $\circ$  the outer product [25].  $U_i$  yields the ratio of the  $i$ th dimension to the principal factors. So, similar to SVD, PARAFAC derives hidden dependencies related to the  $pf$ s and expresses the dimensions of the tensor by relations to the  $pf$ s. Depending on the number of  $pf$ s PARAFAC decomposition can be loss-free. For a third-mode-tensor  $\mathbf{T} \in \mathbf{R}^{k \times l \times m}$  a weak upper bound for this rank is known:  $rank(\mathbf{T}) \leq \min\{kl, lm, km\}$  [25]. There is no proper way for estimating the optimal number of  $pf$ s for an appropriate decomposition but several indicators like residue analysis or core consistency exist [2].



(a) Matrix Decomposition [25]

(b) Tensor Decomposition [25]

The PARAFAC decomposition of a tensor derives authority and hub scores plus additional scores for the relevance of link types (cf. Fig. 2(b)). The tensor  $\mathbf{T}$  in section 3.1 combines information about who loves and hates who. So the PARAFAC decomposition would yield  $U_1$  with subject- $pf$  relation,  $U_2$  with object- $pf$  relation and  $U_3$  with property- $pf$  relation. In other words  $U_1$  keeps the hub scores as relevance of the subjects to the  $pf$ s,  $U_2$  the authorities scores as relevance of the objects to the  $pf$ s and  $U_3$  scores of the relevance of property types to the  $pf$ s. In line with HITS the largest entry of  $U_1^1$  corresponds to the best hub for the first  $pf$  and the largest entry of  $U_2^1$  to the best authority.

By the three matrices  $U_1$ ,  $U_2$  and  $U_3$ , subjects, objects, and properties can be compared in pairs regarding the  $pf$ s derived from all three dimensions. E.g. the relation of a property to specific subjects or objects is derived by multiplying the corresponding column of  $U_3$  with those of  $U_1$  or  $U_2$  respectively.

### 3.3 Ranking Example

Applying the above transformation and analysis to the graph illustrated by Fig. 1 yields the results shown by the first four columns of Table 1. Two groups are identified, one where the predicate hates has a high score, and one where loves is scored highly. The authoritative resources for each group differ from each other. Bob and Chris have high scores with respect to hates. Don and Alex are the top authorities with respect to loves. The application of HITS results in the ranking shown by column 5 and 6. The HITS ranking corresponds to a ranking based on the indegrees of the resources. Notably, the rankings produced by the PARAFAC analysis are different from the HITS results as they provide rankings with respect to different knowledge aspects in the data.

**Table 1.** PARAFAC vs HITS Results

PARAFAC			HITS	
Score	Predicate	Score Resource	Score	Resource
Group 1			0.62	Chris
1.00	hates	0.71 Bob	0.56	Bob
-	-	0.70 Chris	0.50	Alex
Group 2			0.16	Don
1.00	loves	0.70 Don	0.16	Elly
0.001	hates	0.70 Alex		
-	-	0.10 Elly		

## 4 Implementation

Having introduced the theoretical background behind TripleRank, we present the implementation into an applicable system below. We describe the three core components of the TripleRank architecture, which encapsulate a 3-step process, namely i) the collection of data and its transformation to a tensor model, ii) its pre-processing, and iii) analysis.

### 4.1 Data Collection and Transformation

The first process step for the ranking of Semantic Web data is its collection. The TripleRank-collector requests RDF data from (linked open) data providers on the Semantic Web as follows. For a given starting point, i.e. some uniform resource identifier (URI), it executes a breadth first exploration of the surrounding resources. The exploration is parameterized by the maximal exploration depth, the maximal number of statements, and the maximal number of links to follow for each resource and link type. The collected data is then transformed into the tensor representation introduced in Section 3.1.

### 4.2 Pre-Processing

The second component of the TripleRank architecture implements the pre-processing step on the collected data. Pre-processing is applied for two reasons: First, to reduce



the amount of data to be analyzed. Second, to increase the quality of the collected data. Information-theoretic notions ground the pre-processing as implemented by TripleRank. Predicates linking the majority of resources are pruned as they convey little information and *dominate* the data set. More precisely, for all of the results presented in this paper, a threshold of 40% has been used, i.e. predicates that occur in more than 40% of all statements are pruned. The predicate *wikilink* as used within the *dbpedia.org* [4] data set is an example of such a dominating predicate. It corresponds to links between pages of the Wikipedia encyclopedia, e.g. linking from a page describing a music band to associated band members, producers, tracks and further *different* types of information. Accordingly, the semantics of the relation established by this property are rather unclear.

A further pre-processing step is the weighting of the collected data to further remedy the negative effects of domination. We amplify statements based on their predicate frequency so that statements with less frequent predicates are amplified stronger than more common statements. As an effect, the adjacency indicators in the tensor have the following property:

$$\mathbf{T}(x, y, z) = \begin{cases} 1 + \log \frac{\alpha}{\text{links}(z)}, & \text{if } x \text{ points to } y \text{ using property } z \\ 0, & \text{else} \end{cases}$$

The value  $\alpha$  denotes the number of statements in which the most dominant predicate participates. The function  $\text{links}(v)$  ( $\text{links} : V \rightarrow \mathbb{N}_0$ ) returns the number of statements linked by property  $v$ .

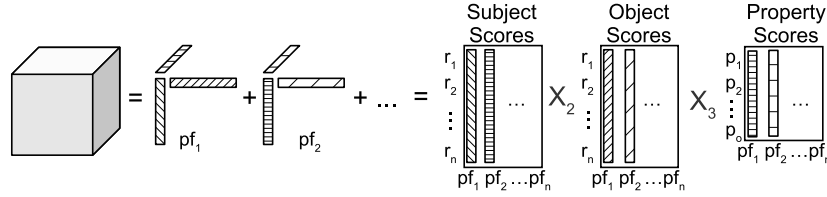
We remark that the implemented pre-processing steps are valuable for generating ranking analyses in general. Notably, simple methods for authority ranking, e.g. the counting of inlink scores per resource and predicate, benefit more from such pre-processing than more complex methods like PARAFAC.

### 4.3 Analysis

The analysis step implements the PARAFAC decomposition of the tensor, as modeled and created by the previous process steps. We have integrated existing software packages [5] for this purpose. As indicated in Section 3.2, the number of factors for the PARAFAC decomposition is crucial for the quality of the results of the analysis. The determination of the optimal number of factors is a case of open research. However, heuristics for determining a suitable number of factors have been published, e.g. the core consistency diagnostic (CORCONDIA) [2]. The factor determination applied in TripleRank builds upon such research.

The result of the analysis is a Kruskal (cf. Sect. 3.2) tensor [25] that approximates the original tensor. As illustrated in Figure 2, the resulting vectors for the first (row), second (column), and third dimension are represented by three matrices. The columns of each of the matrices correspond to the scores calculated for the different factors  $pf_1 \dots pf_n$ . Analogue to the SVD, entries in the column vectors correspond to authority scores, i.e. indicating the relevance of a resource with respect to its in-degree. Entries in the row vectors correspond to hub scores, i.e. indicating the relevance of a resource with respect to its out-degree. We refer to [23] for a thorough analysis of the correspondence





**Fig. 2.** Result of the analysis

between SVD and its interpretation for link analysis. Entries of the vectors in the third dimension indicate the relevance of a resource with respect to the hub and authority resources. Based on this notion, we interpret hub scores as indicative for the relevance of resource when occurring as subject. Vice versa, authority scores indicate the relevance of a resource as object of a statement. As we modeled RDF properties by the third dimension, their relevance can be looked up in the vectors of the third dimension.

## 5 Evaluation

We have applied the implementation of TripleRank to existing Semantic Web data to investigate on the quality of the results and the runtime performance of the approach. Moreover, we have evaluated TripleRank in a user study where we have exploited it to support faceted navigation.

### 5.1 Data Sets

We have created an evaluation testbed by applying the extraction and transformation procedure introduced before. The testbed consists of multiple extracts around resources from dbpedia.org and the SWETO DBLP corpus, e.g. *The Beatles* and *Semantic Web*.

Table 2 describes the different data sets. Data sets created with a different parameterization are separated by a horizontal rule. The table shows, for each data set, the number of statements contained initially, the number of distinct properties, the top 3 properties with respect to the number of statements they occur in (shown in parentheses), and the number of triples after the pre-processing. The first group of data sets has been crawled

**Table 2.** Description of Data Sets

Start Resource	Triples Before	Prop-erties	Top 3 Link Types	Triples After
dbpr:The_Beatles	5084	85	dbpp:wikilink(1719), dbpp:origin(509), dbpp:artist(315)	5084
dbpr:HITS_algorithm	5145	70	dbpp:wikilink(2780), skos:subject(440), dbpp:wikiPageUsesTemplate(143)	5145
dbpr:Berlin	5044	82	dbpp:wikilink(1671), rdf:type(345), dbpp:birthplace(170)	5044
dbpr:The_Lord_of_the_Rings	5035	51	dbpp:wikiPageUsesTemplate(1234), skos:subject(1126), dbpp:wikilink(1063)	5035
dbpr:SPARQL	-	75	skos:subject(2640), rdf:type(426), owl:sameAs(288)	7853
dbpr:James_Bond	18421	93	dbpp:wikilink(13257), rdf:type(182), skos:subject(121)	5164
dbpr:The_Beatles	158608	421	dbpp:wikilink(105254), dbpo:birthPlace(11097), dbpp:countryofbirth(7067)	55354
dblp:semweb/2007	20421	7	rdf:type(9902), dc:publisher(5559), opus:in_series(4849)	10519

Namespaces: dbpr:<http://dbpedia.org/resource/>, dbpp:<http://dbpedia.org/property/>, skos:<http://www.w3.org/2004/02/skos/core#>, dblp:<http://dblp.uni-trier.de/rec/bibtex/>, dc:<http://purl.org/dc/elements/1.1/>, opus:<http://lsdis.cs.uga.edu/projects/semdis/opus#>

by restricting the number of links per resource to 500 and the maximum exploration depth to 10. For the second group, containing the data sets on the SPARQL query language and the James Bond movie, a limit per predicate and resource has been set. For each resource, a maximum of 100 links per link type have been explored while the maximum exploration depth has been set to 8. The final two data sets have been created by restricting only the exploration depth. Here we see significant differences between the data sources dbpedia and dblp. The latter data set contains much less link types compared to the first. Among all<sup>2</sup> dbpedia data sets, we recognize the domination of the predicate wikilink. The crawling parameterization used for the first group of data sets already alleviated the domination, however, for less restrictive crawling parameters there is a strong domination. For instance, statements containing wikilink make up about 72% of all statements in the data set on James Bond. We provide download links for these data sets online at <http://isweb.uni-koblenz.de/Research/DataSets>.

## 5.2 Performance

Applying TripleRank pre-processing and analysis steps to the data sets described above has led to reasonable and interesting results that provide a rich description of the analyzed data sets. As an example, Table 3 shows some of the results for the smaller Beatles data set that contains 5048 triples. As explained above, the PARAFAC analysis yields groupings of predicates and resources. A grouping corresponds to one box in Table 3. The first two columns within a box show for each grouping, the highest ranked predicates including their scores. The third and fourth column show the highest ranked statement objects (authorities) and associated scores. Predicates with a score below 0.1 and resources with a score below 0.0001 have been omitted from the results. For instance, the group on the top left shows the most authoritative resources for the topic described by the predicates skos:subject, dbpo:label, dbpp:label, and dbpp:producer. The top resources in this group are accordingly authoritative for the combination of these predicates and do not necessarily have to be linked by the top predicate, skos:subject for this group. For instance, the resource dbpr:Apple.Records is contained as authority. It describes a music label that produced songs of the Beatles.

As the quality of the results as illustrated by Table 3 seem sensible, we have also measured the runtime performance of the analysis step. The data collection and transformation steps have been excluded as these steps perform linear to the number of statements. Specifically, the performance of the data collection depends heavily on network bandwidth and the performance of the service providing the data. For all of the data sets, the analysis execution times have been within 2 to 18 seconds on a standard laptop computer with a dual core 2.0MHz processor and 2GB RAM.

## 5.3 User Evaluation: Faceted Browsing

We have evaluated TripleRank with 16 test persons to gather objective feedback on the sensibility of its results. The fine-grained results produced by TripleRank can be

<sup>2</sup> The originally crawled data set on SPARQL has been accidentally overwritten with the pre-processed data set so that statements containing wikilink are missing.

**Table 3.** TripleRank Results for the Beatles Data Set

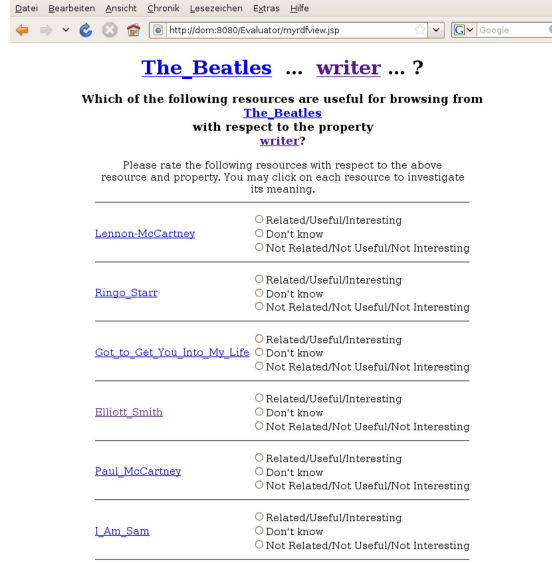
Score	Predicate	Score	Resource	Score	Predicate	Score	Resource
0.66	skos:subject	0.35	dbpr:Category:Songs_produced-by_George_Martin	0.70	dbpo:genre	0.73	dbpr:Rock_and_Roll
0.49	dbpo:label	0.32	dbpr:Category:1968_songs	0.70	dbpp:genre	0.45	dbpr:Beat_music
0.48	dbpp:label	0.27	dbpr:Category:Jazz_songs	0.16	dbpp:wikilink	0.23	dbpr:Psychedelic_rock
0.11	dbpp:producer	0.27	dbpr:Category:Psychedelic_songs	-	-	0.23	dbpr:Jazz_waltz
-	-	0.27	dbpr:Category:Folk_songs	-	-	0.23	dbpr:Folk_music
-	-	0.27	dbpr:Category:The_Beatles_songs-sung_by_George_Harrison	-	-	0.13	dbpr:Folk_rock
-	-	0.27	dbpr:Category:George_Harrison-songs	-	-	0.12	dbpr:Rock_music
-	-	0.16	dbpr:Apple_Records	-	-	0.09	dbpr:And_I_Love_Her
-	-	0.11	dbpr:Category:Apple_Records-albums	-	-	0.06	dbpr:Capitol_Records
-	-	0.11	dbpr:Category:Double_albums	-	-	0.03	dbpr:Country_rock
0.95	dbpp:tracks	0.36	dbpr:Cry_Baby_Cry	0.69	dbpo:label	0.5	dbpr:Apple_Records
0.32	dbpp:wikilink	0.36	dbpr:Long	0.67	dbpp:label	0.26	dbpr:Capitol_Records
-	-	0.16	dbpr:Piggies	0.15	dbpp:wikilink	0.23	dbpr:EMI
-	-	0.16	dbpr:Ob-La-Di	-	-	0.21	dbpo:Resource
-	-	0.16	dbpr:I	-	-	0.18	dbpr:Template:infobox_album
-	-	0.16	dbpr:Rocky_Raccoon	-	-	0.15	dbpo:MusicalWork
-	-	0.16	dbpr:Good_Night	-	-	0.15	dbpo:Work
-	-	0.16	dbpr:While_My_Guitar_Gently-Weeps	-	-	0.14	dbpr:Parlophone
-	-	0.16	dbpr:Mother_Nature	-	-	0.14	dbpr:Template:succession_box
-	-	0.16	dbpr:Julia	-	-	0.1	dbpo:Song
0.96	rdf:type	0.25	yago:TheBeatlesAlbums	0.70	dbpo:recordplace	0.93	dbpr:Abbey_Road_Studios
0.29	skos:subject	0.25	umbel:InstrumentalArtifact	0.70	dbpp:recorded	0.14	dbpr:1966
-	-	0.25	umbel:Artifact	0.14	dbpp:wikilink	0.11	dbpr:9_May
-	-	0.25	yago:AlbumsProducedBy-GeorgeMartin	-	-	0.11	dbpr:19_May
-	-	0.25	dbpo:Album	-	-	0.11	dbpr:16_May
-	-	0.25	yago:Album106591815	-	-	0.09	dbpr:27_April
-	-	0.12	yago:ParlophoneAlbums	-	-	0.09	dbpr:2_June
-	-	0.12	yago:AppleRecordsAlbums	-	-	0.09	dbpr:27_February
-	-	0.12	dbpo:Place	-	-	0.09	dbpr:1_July
-	-	0.12	umbel:Place	-	-	0.09	dbpr:16_July
1.00	dbpp:origin	0.99	dbpr:England				
-	-	0.11	dbpr:Liverpool				

exploited for a number of applications ranging from similarity search for triples to facet identification in RDF data. We have chosen the scenario of faceted browsing for the user evaluation as we believe this scenario is well comprehensible for the participants of the study.

Faceted browsing is a common means to ease the navigation in data by structuring associated data into so called facets [20]. For some resource, e.g. as viewed in a RDF browser, facets correspond to collections of associated resources. Resources within these collections have a commonality with respect to the currently viewed resource, e.g. the facet *band-members* for the resource *The Beatles* should contain only resources describing members of this band, e.g. John Lennon and Paul McCartney.

We have applied TripleRank to automatically select and order resources for given facets. Accordingly, the evaluation has been centered around the following question: *Given a currently viewed resource (subject) and a predicate (facet) associated to it, what are the most interesting, most related, most useful resources (objects) for the facet?*

We have created 10 of such queries using the data sets explained above and presented each of the 10 queries to each of the 16 test persons. As subject and predicate instances we have used the start URIs from the collected data sets (cf. Table 2) and associated properties. Test persons have been presented with candidate resources as pro-



**Fig. 3.** Screenshot of the User Evaluation

duced by the randomly ordered union of the top 10 results from the TripleRank method and a baseline method. For each resource, they could indicate whether it is a good or bad match for the query, or whether they are undecided. Figure 3 illustrates how queries have been presented to test persons by the evaluation tool. Please note that the shown resources (subject, predicate, objects) have been presented as hyperlinks to the service providing the data. Test persons could use these links to investigate on the resource by its complete description. For instance, the resource *The Beatles* links to the web page [http://dbpedia.org/resource/The\\_Beatles](http://dbpedia.org/resource/The_Beatles).

#### 5.4 Baseline Method

The evaluation scenario requires to rank resources with respect to their authority for a given subject and facet. The baseline methods implement a straightforward strategy for computing scores for ranking using the tensor  $\mathbf{T}$  as follows.

**Baseline I** The straightforward and obvious baseline method for computing such a ranking given  $\mathbf{T}$  follows a simple procedure:

1. Select statements matching the subject and predicate (facet) and project the objects.
2. Rank selected objects by their number of inlinks as known from the data set as inlinks indicate authority [23].

For data sets that are available completely and that contain little or no noise, this procedure performs very well with respect to runtime and the quality of results. It will select the relevant objects with high precision. The dbpedia and DBLP data sets obviously contain very little noise. Dbpedia data relies on the careful authoring by numerous users, whereas DBLP data maps to settled and rigid structures maintained by a database system. However, due to the fact that our data sets have been crawled from online services, they are incomplete and represent only a small portion of these large data sets.

Accordingly, when applying this baseline method to our data sets, we have experienced no or very little recall due to the sparsity of the data. For instance, applying this baseline for answering the query “*What are relevant subjects for James Bond?*” it returns only one resource while the enhanced Baseline II method we present in the following returns 6 relevant resources. Nevertheless, we imagine that every method for ranking facet objects will include this baseline method as it produces high precision when data sets are complete and without noise.

**Baseline II** We modified the Baseline I method in order to improve its competitiveness, i.e. improving its recall for incomplete data sets. Instead of considering only exact matches of subject-predicate patterns, we extended this criteria to consider also predicate matches only. As the data sets have been created around a URI that is central within the data sets, this extended baseline method turned out to produce more competitive results, specifically an increased recall. This method is also closer to the PARAFAC analysis as used by TripleRank as it corresponds to the authority ranking based on the in-degree (number of incoming links) of resources. In fact, this method can be seen as an approximation of authority scores in HITS, which are highly correlated with in-degrees of nodes [14] (see also the example given in Sect. 3.3). Accordingly, we have evaluated TripleRank against this improved Baseline II method.

## 5.5 TripleRank Method

The TripleRank method is similar to the Baseline II method with the exception that it builds upon the results of the PARAFAC decomposition of  $\mathbf{T}$  instead of plain link-counting using  $\mathbf{T}$ . First, the top factors for the predicate (facet) in question are selected. They are looked up from the matrix containing the predicate scores (cf. Figure 2). Those factors are selected where the predicate is ranked first or has a score of at least 0.6. Afterwards, the resource vectors of these factors, as given by the object-score matrix, are considered for the selection of the ten resources with the highest scores.

## 5.6 Results

We have received 1387 answers by 16 test persons. On average, the union of the results produced by the BaselineII and TripleRank method led to 8.669 candidate objects per query. While we asked for the top 10 results by each method, the average number of results is lower, as for several questions both methods produced less than 10 results.

We calculated the agreement among test persons using the index raw agreement [17]. It represents the ratio between the measured inter-rater agreement and the best possible agreement. Accordingly, a value of 1 corresponds to the maximal agreement, while 0 corresponds to no agreement. As expected, the evaluation shows a high inter-rater agreement. Considering the specific agreement on the relevance of results, test persons showed a higher agreement rate on positive candidates (agreement-ratio: 0.68) than on negative candidates (agreement-ratio: 0.57). For the overall agreement, we have measured an even higher ratio of 0.70 among the test persons.

The evaluation results presented in Table 4 built on these user ratings and give insight into the constitution of the union of the results. Table 4 lists the average number of results found exclusively by each method (BaselineII–TripleRank, TripleRank

–BaselineII), and the average number of results suggested by both of the methods, namely the intersection of their results ( $\text{BaselineII} \cap \text{TripleRank}$ ). In addition, Table 4

**Table 4.** Results of the User Evaluation

	Avg Number of Results				Avg Precision	
	Total	Positive	Negative	Undecided	Macro	Micro
BaselineII–TripleRank	1.075	0.273	0.444	0.344	0.478	0.393
TripleRank–BaselineII	4.731	1.913	1.650	1.169	0.521	0.537
BaselineII $\cap$ TripleRank	2.863	1.338	0.763	0.763	0.659	0.637
BaselineII	3.948	1.626	1.207	1.107	0.607	0.574
TripleRank	7.594	3.251	2.413	1.932	0.557	0.574

also presents the average numbers of ratings received for the results. Here, we notice that the TripleRank method succeeds in suggesting more positively rated results than negative ones. Comparing the total number of results produced by each method, the TripleRank approach nearly doubles the number of results (3.948 vs 7.594). For the number of results considered relevant by users, TripleRank achieves an increase from 1.626 to 3.251, i.e. the recall is increased by the factor of 1.999. For these results, we have also calculated additional confidence intervals with confidence degree 0.9. In most of the cases, TripleRank outperforms the baseline with statistical significance.

Table 4 also lists the precision of each approach. For a fine-grained analysis, we have calculated the macro precision, reflecting the users’ perspective on the performance of the methods, and the micro precision, reflecting the systems view [11]. For the results produced exclusively by each method, TripleRank has higher precision than the baseline approach. Considering the overall results, the baseline method has equal precision on the micro-level and higher precision on the macro-level. Significance tests with confidence degree 0.9, however, have shown that all measured differences in precision are not statistically significant. Accordingly, we conclude that the TripleRank approach results in a substantially increased recall without loss of precision.

## 5.7 Lessons Learned: TripleRank Advantages

By further analysis of the evaluation results, we have investigated why the TripleRank approach outperforms the baseline method. Among the findings, we identified that the results of the analysis step implicate similarities among properties. For instance, it turned out that the dbpedia data sets contain predicates with similar semantics, e.g. the predicate <http://dbpedia.org/property/genre> and the predicate <http://dbpedia.org/ontology/genre> (cf. Table 3). Both properties associate bands or songs to a genre. The analysis step identified this similarity producing a factor where both properties are among the top two properties. The most authoritative resources for that factor are accordingly those resources having both or either one of these properties as inlink. As an effect, both precision and recall are increased when compared to baseline approaches that cannot identify the similarities of links. The advantage of TripleRank, however, is not only given when properties are directly grouped

into one factor. The results of the tensor analysis also enable to increase recall by considering multiple factors where a predicate is ranked high. Within our experiments such increase of recall has been achieved without a loss of precision.

## 6 Conclusion and Future Work

In this paper we presented TripleRank, a novel approach for authority ranking in Semantic Web applications. Conceptually, TripleRank is a Semantic Web correspondent to authority ranking methods known from Web retrieval, such as PageRank or HITS. Our approach exploits the novel representational model for semantic RDF graphs, based on 3-dimensional tensors. This allows us to exploit in the natural way the available semantics of RDF predicates. By applying the PARAFAC tensor decomposition we identify authoritative sources in the knowledge base as well as groups of semantically coherent predicates and resources. The systematic evaluation shows the viability of TripleRank for a wide range of search/browsing applications in the Semantic Web, such as semantic faceted navigation and ranked retrieval. Therefore, TripleRank can be seen as a next step towards efficient and effective search/retrieval technology for Semantic Web.

As next steps for this research, we pursue the integration of TripleRank into existing Linked Open Data browsers, e.g. our own Semantic Web browser LENA [24], to investigate on further enhancements of the method. We also envision the addition of semantic relations that go beyond link semantics. For instance, we consider the modeling of resource similarity based on the similarity of the types of resources.

*Acknowledgements* This work was funded by the X-Media project ([www.x-media-project.org](http://www.x-media-project.org)) sponsored by the European Commission as part of the Information Society Technologies (IST) programme under EC grant number IST-FP6-026978.

## References

1. Boanerges Aleman-Meza, Christian Halaschek-Wiener, Ismailcem Budak Arpinar, Cartic Ramakrishnan, and Amit P. Sheth. Ranking complex relationships on the semantic web. *IEEE Internet Computing*, 9(3):37–44, 2005.
2. Claus A. Andersson and Rasmus Bro. The n-way toolbox for matlab. *Chemometrics and Intelligent Laboratory Systems*, 52(1):1 – 4, 2000.
3. Kemafor Anyanwu and Amit P. Sheth. The p operator: Discovering and ranking associations on the semantic web. *SIGMOD Record*, 31(4):42–47, 2002.
4. Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary G. Ives. Dbpedia: A nucleus for a web of open data. In *ISWC/ASWC*, pages 722–735, 2007.
5. Brett W. Bader and Tamara G. Kolda. Algorithm 862: MATLAB tensor classes for fast algorithm prototyping. *ACM Transactions on Mathematical Software*, 32(4):635–653, December 2006.
6. Andrey Balmin, Vagelis Hristidis, and Yannis Papakonstantinou. Objectrank: Authority-based keyword search in databases. In *VLDB*, pages 564–575, 2004.
7. Krishna Bharat and Monika Rauch Henzinger. Improved Algorithms for Topic Distillation in a Hyperlinked Environment. In *21st Annual International ACM SIGIR Conference, Melbourne, Australia*, pages 104–111, 1998.
8. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Seventh International World-Wide Web Conference (WWW 1998)*, 1998.



9. Andrei Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.
10. Paul Alexandru Chirita, Stefania Ghita, Wolfgang Nejdl, and Raluca Paiu. Beagle++ : Semantically enhanced searching and ranking on the desktop. In *ESWC*, 2006.
11. Hinrich Schuetze Christopher Manning, Prabhakar Raghavan. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
12. David A. Cohn and Thomas Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. In *13th Conference on Advances in Neural Information Processing Systems (NIPS)*, Denver, USA, pages 430–436, 2000.
13. Michelangelo Diligenti, Marco Gori, and Marco Maggini. Web Page Scoring Systems for Horizontal and Vertical Search. In *11th International World Wide Web Conference (WWW)*, Honolulu, USA, pages 508–516, 2002.
14. Chris H. Q. Ding, Xiaofeng He, Parry Husbands, Hongyuan Zha, and Horst D. Simon. PageRank, HITS and a Unified Framework for Link Analysis. In *25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland*, pages 353–354, 2002.
15. Li Ding, Rong Pan, Timothy W. Finin, Anupam Joshi, Yun Peng, and Pranam Kolari. Finding and ranking knowledge on the semantic web. In *International Semantic Web Conference*, pages 156–170, 2005.
16. Magdalini Eirinaki and Michalis Vazirgiannis. Usage-based pagerank for web personalization. *Data Mining, IEEE International Conference on*, 0:130–137, 2005.
17. Alexander Von Eye and Eun Young Mun. *Analyzing Rater Agreement: Manifest Variable Methods*. Lawrence Erlbaum Associates, 2004.
18. Richard A. Harshman and Margaret E. Lundy. Parafac: Parallel factor analysis. *Computational Statistics & Data Analysis*, 18(1):39–72, August 1994.
19. Taher H. Haveliwala. Topic-sensitive PageRank. In *11th International World Wide Web Conference (WWW)*, Honolulu, USA, pages 517–526, 2002.
20. Michiel Hildebrand, Jacco van Ossenbruggen, and Lynda Hardman. /facet: A browser for heterogeneous semantic web repositories. In *International Semantic Web Conference*, pages 272–285, 2006.
21. Aidan Hogan, Andreas Harth, and Stefan Decker. Reconrank: A scalable ranking method for semantic web data with context. In *2nd Workshop on Scalable Semantic Web Knowledge Base Systems*, 2006.
22. Glen Jeh and Jennifer Widom. Scaling Personalized Web Search. In *12th International World Wide Web Conference (WWW)*, Budapest, Hungary, pages 271–279, 2003.
23. Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.
24. Jörg Koch, Thomas Franz, and Steffen Staab. Lena - browsing rdf data more complex than foaf. In *International Semantic Web Conference (ISWC) Demo Session*, 2008.
25. Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3), September 2009 (to appear).
26. Tamara G. Kolda, Brett W. Bader, and Joseph P. Kenny. Higher-Order Web Link Analysis Using Multilinear Algebra. In *5th IEEE International Conference on Data Mining (ICDM)*, Houston, USA, pages 242–249, 2005.
27. Ronny Lempel and Shlomo Moran. SALSA: the Stochastic Approach for Link-Structure Analysis. *ACM Transactions on Information Systems (TOIS)*, 19(2):131–160, 2001.
28. Yu-Ting Liu, Bin Gao, Tie-Yan Liu, Ying Zhang, Zhiming Ma, Shuyuan He, and Hang Li. Browserank: letting web users vote for page importance. In *SIGIR*, pages 451–458, 2008.
29. Davood Rafiei and Alberto O. Mendelzon. What is this Page known for? Computing Web Page Reputations. *Computer Networks*, 33(1-6):823–835, 2000.
30. Cartic Ramakrishnan, William H. Milnor, Matthew Perry, and Amit P. Sheth. Discovering informative connection subgraphs in multi-relational graphs. *SIGKDD Explor. Newsl.*, 7(2):56–63, 2005.
31. Matthew Richardson and Pedro Domingos. The Intelligent surfer: Probabilistic Combination of Link and Content Information in PageRank. In *14th Conference on Advances in Neural Information Processing Systems (NIPS)*, Vancouver, Canada, pages 1441–1448, 2001.
32. Cristiano Rocha, Daniel Schwabe, and Marcus Poggi de Aragão. A hybrid approach for searching in the semantic web. In *WWW*, pages 374–383, 2004.