AN RDF METADATA-BASED WEIGHTED SEMANTIC PAGERANK ALGORITHM

Hee-Gook Jun[1], Dong-Hyuk Im[2], and Hyoung-Joo Kim[3]

[1,3]School of Computer Science and Engineering Seoul National University
[2]Dept. of Computer and Information Engineering, Hoseo University

ABSTRACT

PageRank evaluates the importance of Web pages with link relations. However, there is no direct method of evaluating the meaning of links in a hyperlink-based Web structure. This feature may cause problems in that pages containing many in-links are highly ranked without considering the meaning of the link relations among the pages. We therefore propose a novel ranking approach to directly analyze the meaning of links by transforming a hyperlink-based Web structure into a semantic-link-based Web structure. We extract semantic metadata from Web pages and construct a semantic-link-based Web structure using RDF model. We define a metric to evaluate the weight of the links for stratifying rank values based on their importance in the semantic-link-based Web structure. We implement the weighted semantic ranking algorithm in the MapReduce framework to consider large-scale semantic metadata. The results of our experiment show that our approach outperforms existing PageRank algorithms.

KEYWORDS

Semantic Web, RDF, RDFa, PageRank, Big Data, MapReduce.

1. INTRODUCTION

PageRank is a representative link-based ranking method [1, 2] in modern Web information retrieval [3–5]. The authors of PageRank assume that pages with many in-links from other pages are important. In PageRank, each page distributes its rank value to other pages through links among the pages. However, the page rank values are equally distributed without considering the meaning of the links. Due to this feature, unimportant pages containing many in-links could be highly ranked.

Further studies have tried to improve the rank value propagation [6–10], and some have considered link evaluation. Improved methods using link weighting cause pages containing many meaningless in-links to be low ranked. However, in the current Web structure, links between pages are defined by hyperlink notation. Because hyperlink notation cannot express the meaning of links, it is not easy to directly evaluate the weight of the links. Therefore, existing works have had to use indirect methods to evaluate the link weights. Moreover, simply analyzing links is insufficient to measure the importance of pages, in that highly ranked pages containing many in-links are not always important, and may even contain meaningless information.

In this paper, we propose the Weighted Semantic PageRank (WSPR), which evaluates links directly to obtain a more accurate ranking result. We utilize semantic information for WSPR, and form a semantic-link-based Web structure to manage the semantic information. A semantic-

link-based Web structure is created from a hyperlink-based Web structure by using RDF [11] metadata. Links in a semantic-link-based Web structure contain semantic information and are used to resolve the problem of determining the weight of the links. Thus, we are able to calculate rank values based on the semantic information analyzed from pages and links. Furthermore, WSPR reduces the phenomenon of giving high ranks to unimportant pages. As WSPR uses semantic resources in pages rather than meaningless hyper-links to calculate rank values, it enables important pages to receive high rank values. In addition, we have implemented the WSPR algorithm using the Hadoop framework [12] to manage large-scale semantic metadata more effectively.

The contribution of this paper is threefold. First, we propose a ranking method based on semantic information to generate more accurate ranking results. Our method considers the meaning of pages and links by evaluating their semantic information in a semantic-link-based Web structure, rather than using the number of links among the pages. Second, we design an algorithm to reduce the probability of giving high rank values to unimportant pages. Using semantic information instead of hyper-links, the method is able to calculate rank values based on the page meaning. Thus, the method guarantees that highly ranked pages contain valuable information. Finally, we implement a framework that transforms a hyperlink-based Web structure into a semantic-link-based Web structure, and evaluates the page rank values based on this structure. The system runs on the Hadoop framework to manage large scale semantic metadata.

This paper has been extended from previous work [13] that discussed the method aspect of utilizing RDF metadata in the ranking process. The present work represents our latest results and describes an additional framework to build semantic-link-based Web structures.

The remainder of this paper is structured as follows. In the following section, we provide an overview of PageRank algorithms in view of evaluating the link weights. In Section 3, we introduce a semantic-link-based Web structure and provide our framework to construct the structure. In Section 4, we present our ranking method in detail. Section 5 reports the results of our experiments. Finally, we present our conclusions and our perspectives for future work in Section 6.

2. RELATED WORK

Search engines answer user queries with ranked page lists created using ranking methods. Early ranking methods were term-based ranking methods that evaluate page importance based on the number of matched terms for a given query [14, 15]. After 1998, alternative link-based ranking methods were provided and demonstrated much higher performance than term-based ranking methods. PageRank [1] and HITS [2] are representative link-based ranking methods. While HITS considers both in-links and out-links to classify pages into authority and hub, PageRank only considers in-links focusing on ranking pages by their popularity. PageRank calculates the rank score as follows:

$$PR(r_i) = d \sum_{j \to i} \frac{1}{N_j} \cdot PR(r_j) + (1 - d), \qquad (1)$$

where d is a damping factor to reflect user behavior. The damping factor is a probability value, which is usually set to 0.85 because PageRank assumes that users have an 85% probability of following the link chain, and a 15% probability of jumping to a new page. In equation 1, the PageRank value of a page is the sum of the PageRank values of pages that refer to this page. Each page equally propagates its rank value to related pages.

Figure 1 shows an example of PageRank propagation. Page A with rank value 30 assigns a PageRank of 15 to pages B and C. Similarly, page D assigns a PageRank of 20 to B and C. However, a problem arises from the fact that the rank score of the previous page is equally distributed, without considering the meaning of the links. This feature may cause meaningless pages with many in-links to be highly ranked in the search lists.

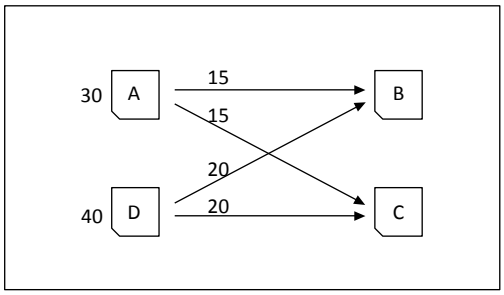


Figure 1. PageRank example.

Weighted PageRank [7] is an alternative method to avoid the uniform rank value distribution without considering the meaning of links. Weighted PageRank stratifies the distribution of the rank values based on the link weights (Figure 2). Equation 2 indicates that link weights are calculated by the proportion of the number of in-links and out-links. However, this method still considers the number of links recursively to evaluate the weight of a link. Furthermore, because the method estimates the importance of pages by using hyper-links, it does not always guarantee that a page contains information relevant to a user's query.

$$W^{in}_{(v,u)} = \frac{I_u}{\sum_{p \in R(v)} I_p} \quad , \quad W^{out}_{(v,u)} = \frac{O_u}{\sum_{p \in R(v)} O_p} \tag{2}$$

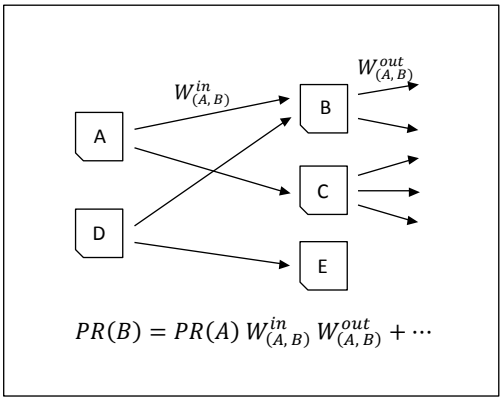


Figure 2. Weighted PageRank example.

Since Weighted PageRank, many approaches have been developed to more precisely evaluate the weight of the links. Weighted Page Content Rank [8] uses Web content mining to improve the Weighted PageRank. Weighted Page Content Rank utilizes not only the weight of the links but also the correlation between user queries and search results. However, the computation of Weighted PageRank is still based on the number of links and does not take into account the semantic meaning of the links. Other methods, such as Topic-Sensitive PageRank [9] and personalized PageRank [10], utilize additional information. Topic-Sensitive PageRank classifies Web pages according to their topics, and computes rank values by applying a query-biased metric on the set of classified pages. Personalized PageRank is a user-biased metric that

provides specific search results for each individual user. On the other hand, our primary goal is to build an integrated ranking algorithm as well as utilize semantic metadata. Thus, we set the scope of our research to an unbiased and explicit semantic analysis of page ranking.

3. SEMANTIC-LINK-BASED WEB STRUCTURE

3.1 Background

A semantic markup language is used to embed metadata in Web documents. RDFa [16], Microformats [17], and Microdata [18] are representative semantic markup languages. As RDFa, derived from the RDF data model, is a W3C Recommendation and provides high applicability in the Web environment, we mainly focus on RDFa to build a semantic-link-based Web structure. RDFa enables Web documents to contain semantic metadata in RDF. The RDF data model is a language to represent conceptual models. RDF data consists of triples comprising a subject, predicate, and object. A subject and an object are linked by a predicate in a triple (Figure 3), and a set of triples forms a directed graph. Figure 4 shows an example of using RDFa to annotate RDF metadata in XHTML code. Web documents written in XHTML code with RDFa are viewed as Web pages in Web browsers; moreover, the documents are used as semantic metadata by adopting RDFa parsers. This feature provides a basis for the creation of a semantic-link-based Web structure.

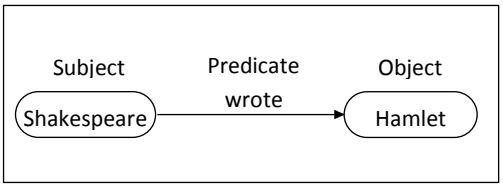


Figure 3. Example RDF triple.

XHTML with RDFa

http://.../Example.html

```
<div about="http://.../GreatAdventure">
    <h3 property="dc:creator">John</h3>
</div>
```

Figure 4. Example RDFa annotation.

Semantic metadata management has been developed in various fields. Google, Microsoft, and Yahoo! established schema.org in 2011 for precise search results using RDFa, Microformats, and Microdata. Google's Rich Snippet and Yahoo!'s BOSS (Build your Own Search Service) are technology that use semantic metadata in search results [19]. Facebook's Open Graph protocol is for semantic metadata in a social network [20]. Drupal and Wordpress, major content management systems, provide an automatic semantic tagging module [21]. Various RDFa-related methods as well as RDFa annotation systems [22–25] have also been developed. W3C has provided a distiller and a parser for RDFa. RDFauthor [26] is an integrative RDFa management framework.

3.2 Semi-automatic RDFa Annotation System

Our ranking method operates on a semantic-link-based Web structure to obtain semantic input data. A semantic-link-based Web structure is constructed from a hyperlink-based Web structure to use of semantic metadata on the Web. There are three kinds of methods to build a semantic-link-based Web structure. Semantic metadata already embedded in Web pages can be obtained by parsers. An extractive method is used when Web pages do not contain semantic metadata but contain semi-structured metadata, such as table data. However, the preceding methods do not process Web pages without any metadata. Thus, we have developed a semi-automatic RDFa annotation system (Figure 5). The system receives Web pages as input and matches the pages to semantic resources of linked open data. The system also provides a method to manually annotate semantic metadata. Finally, the system generates semantic data-annotated Web pages as output.

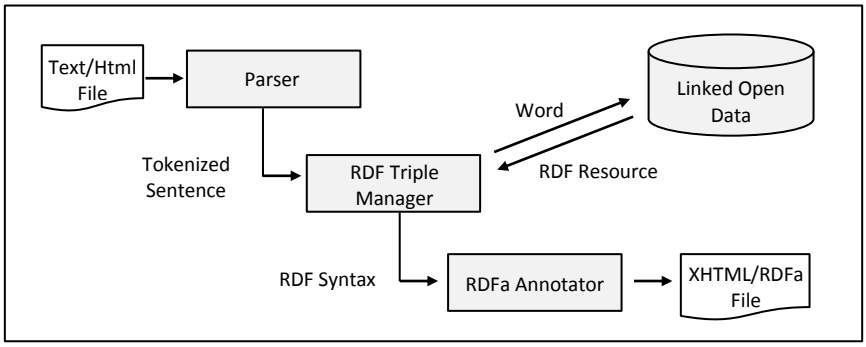
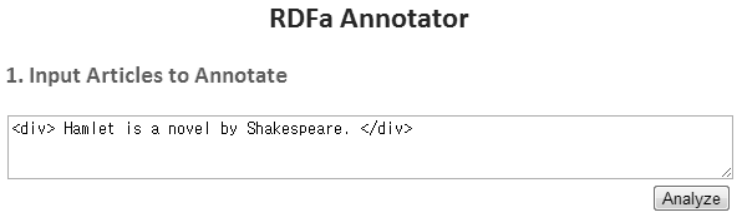


Figure 5. Semi-automatic RDFa annotation system.

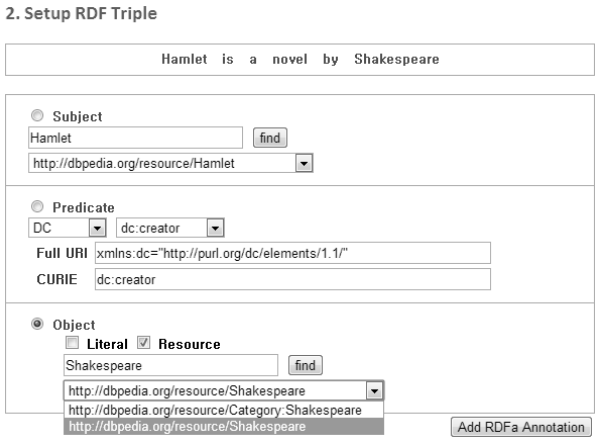
The semi-automatic RDFa annotation system includes the following components:
Input Box: The system reads the input data, parses the data sentence by sentence, and tokenizes each sentence into words (Figure 6(a)). Then, the parsed data is moved to an RDF triple management procedure.

RDF Triple Manager: The system finds candidate resources that match the words from Linked Open Data, such as DBPedia [27]. The user can then choose an appropriate resource from among the candidates, as shown in Figure 6 (b). The user places words in the input box as the subject of a triple. In addition, the system provides a set of predicates from RDF vocabularies for interoperability.

Result Code View: The system generates RDFa meta tags based on the information provided by the previous step. Finally, the resource code view shows a Web document with RDFa annotations.



(a) Input box



(b) RDF triple manager

3. Result RDFa Codes

```
<html
 xmlns="http://www.w3.org/1999/xhtml"
 xmlns:dc="http://purl.org/dc/elements/1.1/"
 xmlns:dbpedia="http://dbpedia.org/resource/"
 >

  <span about="dbpedia:Hamlet">
   <span rel="dc:creator">
    <span about="dbpedia:Shakespeare"></span>
   </span>
  </span>
```

(c) Code view results

Figure 6. User interface of the provided annotation system.

4. WEIGHTED SEMANTIC PAGERANK

4.1 Proposed Architecture

Our ranking method, called Weighted Semantic PageRank (WSPR), provides a novel page importance computation method based on a semantic-link-based Web structure. WSPR directly computes the weight of the links by evaluating their meaning. WSPR is composed of four procedures (Figure 7). The first two procedures are related to the transformation of a hyperlink-based Web structure to a semantic-link-based Web structure. The other procedures compute rank values based on the semantic-link-based Web structure built in the previous procedures.

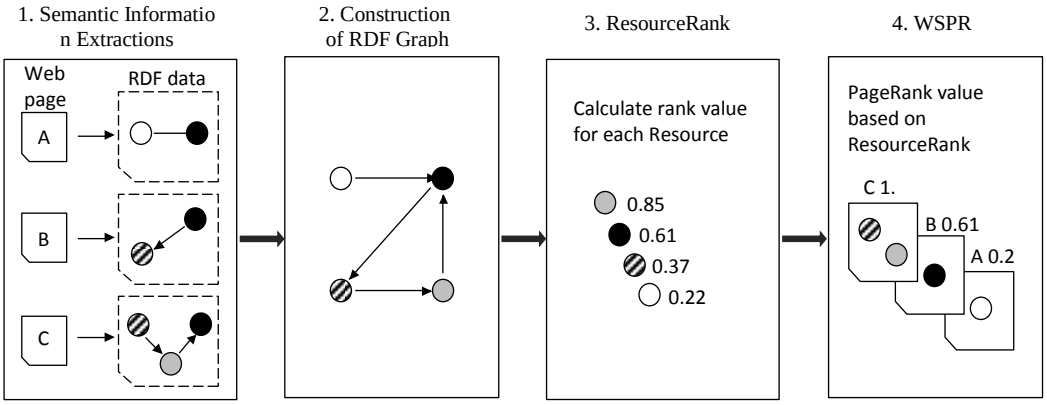


Figure 7. Overview of WSPR system steps.

4.1.1 Semantic Information Extraction

In the first process of WSPR, the system extracts RDF metadata from the pages. Before starting the first process, we utilized the semantic metadata annotation method mentioned in the previous section for more effective execution of the process. Then the system collects semantic metadata from Web pages (Figure 8). In WSPR, collected RDF resources are used as the unit of rank values, and predicates between resources are viewed as labeled links to determine the resources' degree of importance.

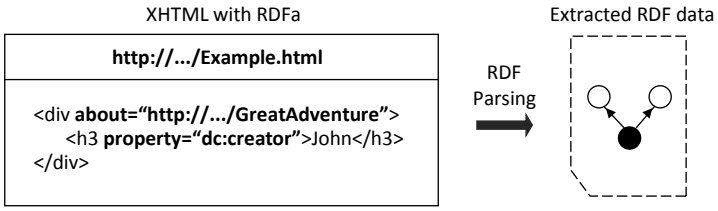


Figure 8. RDF parsing of an RDFa-annotated Web page.

4.1.2 Construction of an RDF Graph

In the second procedure, WSPR integrates the RDF dataset from the previous process into a single structure. To interconnect multiple RDF datasets, the system matches resources with a Uniform Resource Identifier (URI), which is a unique value that identifies resources on the Web. Figure 9 shows an example of RDF data integration. The system finds resources with the same URI (the black nodes in Figure 9), and joins the matched resources into one resource. After the procedure, all resources are connected with one another, based on the URI. The combined graph is viewed as a semantic-link-based Web structure to be used for the evaluation of semantic resource ranks in the next procedure.

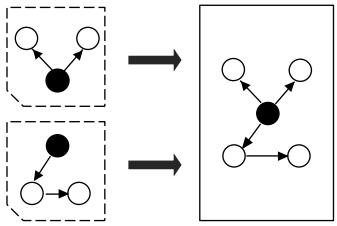


Figure 9. Merging RDF triples with resources having the same URI.

4.1.3 ResourceRank

In the third procedure, WSPR begins the rank evaluation of resources using ResourceRank, which calculates the rank values using the semantic-link-based Web structure created in the second procedure. It then evaluates the weight of the links using semantically labeled predicates among the resources. The calculated weight values stratify the distribution of resource rank values, based on the degree of their semantic relationships.

The weight of the links is calculated either manually or automatically [28]; we focused on generating an automatic link-weight computation metric. WSPR evaluates predicates as links in a semantic-link-based Web structure. First, WSPR calculates the Predicate Frequency (PF) of the semantic-link-based Web structure (Equation 3). The Predicate Frequency uses a function f that returns the raw frequency of a predicate. The Predicate Frequency is also normalized, by dividing by the maximum raw predicate frequency in the resource. Moreover, WSPR uses the Inverse Predicate Frequency (IPF) for balancing Predicate Frequency values.

$$PF(p,r) = \frac{f(p,r)}{\max\{f(w,r): w \in r\}} \; and \tag{3}$$

$$IPF(p,R) = log \frac{|R|}{|\{r \in R: p \in r\}|}, \tag{4}$$

where p is a target predicate to compute the weight, r is a resource, and R is a set of resources. Then, the weight of the links is defined by Equation 5:

$$weight(r_i,p) = PF(r_i,p) \times IPF(r_i,p). \tag{5}$$

Finally, the ResourceRank equation takes on the form,

$$RR(r_i) = d \sum_{j \in outlink(i)} \frac{RR(r_j) \cdot weight(r_j,p)}{\sum_{j \in outlink(i)} weight(r_j,p)} + (1-d), \tag{6}$$

where $RR(r_j)$ is the ResourceRank value of a resource linked to resource r_i. $RR(r_j)$ is stratified based on its weight value before being added to $RR(r_i)$.

4.1.4 Weighted Semantic PageRank

The final procedure of WSPR is the computation of page rank values. The page rank values are calculated based on the resource rank values from the previous procedure. Resource rank values are returned to the pages that respectively contain the resources. It means that the resource importance is used to evaluate the importance of the pages that contain the resources. In other words, the reputation of a page is measured by the importance of the semantic resources that the page contains, rather than the number of links in the page. It improves the limitations of the link-weight evaluation methods, based on the number of links. It also guarantees that a page with higher rank values always contains semantically important information. Thus, the probability that meaningless pages will receive high rank values is lower than in previous approaches.

Equation 7 shows the equation for the PageRank score, based on the ResourceRank scores computed in the previous procedure.

$$WSPR(p_i) = \sum_{r \in p_i} RR(r), \tag{7}$$

where $RR(r)$ is the ResourceRank value of resource r, which is contained in page p_i. Finally, the WSPR value of page p_i is the summation of all ResourceRank values of resources in page p_i.

4.2 MapReduce Algorithm

The Hadoop framework [29] is an open source implementation of Google's MapReduce framework. Hadoop provides the Hadoop Distributed File System (HDFS) that distributes and manages large datasets over multiple servers. Hadoop MapReduce is used to perform parallel computations on Hadoop clusters. Using the MapReduce framework, researchers are able to concentrate more on implementing their algorithm and less on the parallel processing elements.
A MapReduce job consists of two components, a mapper and a reducer (Figure 10). In the map phase, input data is converted into key-value pairs. The key-value pairs are sent to the reduce phase by keys. In the reduce phase, the output dataset is generated by applying computations to the pairs received from the map phase. The MapReduce algorithm is based on the concept of map/reduce in functional programming. The method is simple and powerful, as the function runs with fault tolerant feature on parallel and distributed systems.

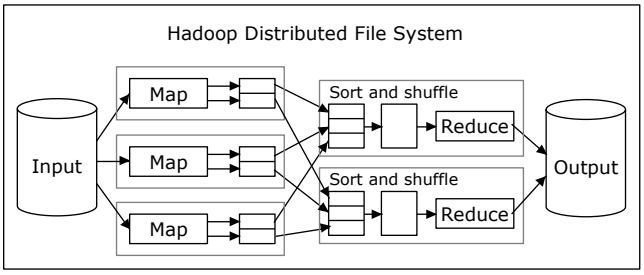


Figure 10. Overview of Hadoop MapReduce.

WSPR is implemented on Hadoop to account for the large-scale semantic metadata. The WSPR MapReduce algorithm consists of three jobs (Figure 11). The first job receives Web pages with semantic information as input data, and calculates the ResourceRank values of resources in the input data. After calculating the ResourceRank values, the first job outputs the result to the next job (Figure 12).

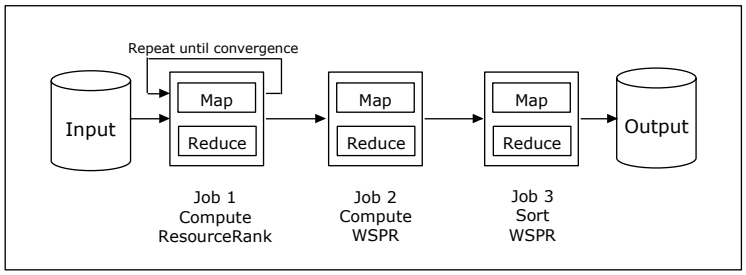


Figure 11. WSPR MapReduce job framework.

The second job receives the RDF resource information with ResourceRank values to compute the WSPR value. The ResourceRank scores of resources are assigned to each page where the resources were originally contained. The WSPR value of a page is calculated by summing the ResourceRank values assigned to the page (Figure 13).

The intermediate ranking result from the second job is sent to the third job to be ordered by WSPR values. Finally, the third job outputs the page ranking result, based on the WSPR values (Figure 14).

```
class MAPPER
  method MAP(pageid i, page P)
    EMIT(pageid i, page P)   // Emit adjacency list
    for all pageid j ∈ P.AdjacencyList do
      r ← j.ResourceRank × j.LinkWeight
      EMIT(pageid j, r)        // Emit value for ResourceRank
    end

class REDUCER
  method REDUCE(pageid i, values [v1, v2, …])
    R ← ∅
    sum ← 0
    for all v ∈ values [v1, v2, …] do
      if IsResourceRankScore(v) then
        sum ← sum + v        // Sum of values for ResourceRank
      else
        R.AdjacencyList ← v   // Get adjacency list information
      end
    end
    R.ResourceRank ← sum × 0.85 + 0.15  // Compute rank
    EMIT(pageid i, page R)
```

Figure 12. MapReduce Job 1: ResourceRank.

```
class MAPPER
  method MAP(pageid i, page P)
    EMIT(pageid i, P.resourceRank)

class REDUCER
  method REDUCE(pageid i, resourceRanks [r1, r2, …])
    R ← ∅
    sum ← 0
    for all r ∈ resourceRanks [r1, r2, …] do
      sum ← sum + r        // ResourceRank value summation
    end
    R.PageRank ← sum
    EMIT(pageid i, page R)
```

Figure 13. MapReduce Job 2: WSPR.

```
class MAPPER
  method MAP(pageid i, page P)
    EMIT(P.PageRank, pageid i)  // Sort using Reduce function
```

Figure 14. MapReduce Job 3: Ordering page by rank score.

5. EXPERIMENTAL EVALUATION

5.1 System Setup

We performed all experiments on twelve nodes in our cluster. One node served as the master node, while the other 11 were slave nodes. Each node had a 3.1 GHz quad-core CPU, 4GB memory, and 2TB hard disk. The operating system was 32-bit Ubuntu 12.04.2. We used Hadoop version 1.2.1 running on Java 1.6.0

We used 80,000 Wikipedia [30] web pages as a source of Web data, and extracted 500,000 RDF metadata from infobox tables in the Wikipedia pages.

5.2 Results

In Figures 15 to 17, we present the results for the ranking methods on the uniform page dataset. Each graph corresponds to the page ranking performance. The horizontal axis represents the size of the page set. The vertical axis in each graph is the precision, recall, or f-measure, respectively. Our enhanced method consistently improves performance beyond the other methods (PageRank, Weighted PageRank, and Topic-Sensitive PageRank). It shows that WSPR provides fewer false positive and false negative ranking results.

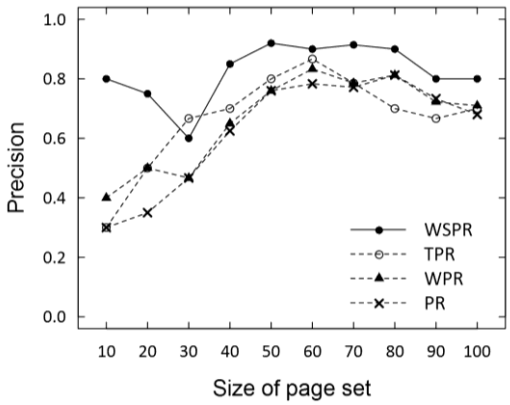


Figure 15. Precision of page rank methods for a varying number of pages.

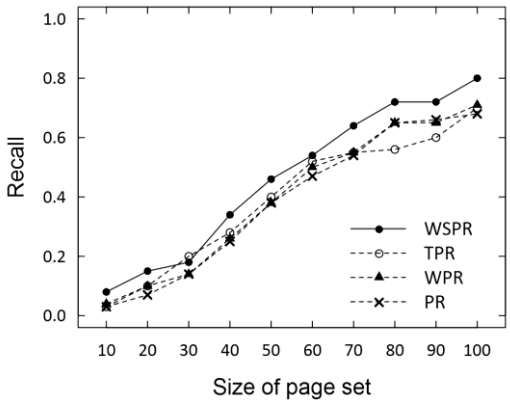


Figure 16. Recall of page rank methods for a varying number of pages.

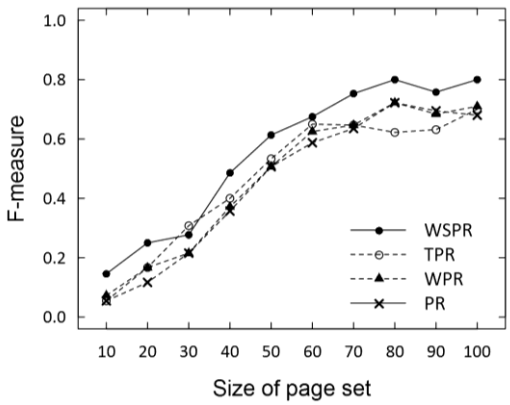


Figure 17. F-measure of page rank methods for a varying number of pages.

We also used the Normalized Discounted Cumulative Gain (NDCG) [31] to measure the effectiveness of the results provided by ranking methods. Table 1 shows the NDCG at each rank position k of the ranking methods. PR, which distributes the rank values uniformly, is the lowest. The other two methods that distribute rank values using the link weights generally outperform PR. WSPR, which evaluates the weight of the links semantically and ranks pages based on the importance of their semantic resources, achieves the highest ranking accuracy in terms of the relevance of the results to the search goals.

Table 1. NDCG@k results for the test query.

NDCG@k	PR	WPR	TPR	WSPR
NDCG@5	0.8765	0.9838	0.9854	0.9931
NDCG@8	0.8824	0.9469	0.9605	0.9748
NDCG@10	0.8866	0.9389	0.9563	0.9732

Table 2 shows a more detailed view of the ranking results for a query about literature. In the process of ResourceRank computation, WSPR extracted two resources, "Macmillan" and "Publishing company," from the page written about "Macmillan." The ResourceRank values for the two resources are 1.118 and 0.429, respectively. In addition, WSPR extracted one resource, "United State," with ResourceRank value 1.272 from the page written about "United States." Although the ResourceRank value of "United State" is higher than those of the other resources, the WSPR value for "United States" is lower than the other page's WSPR value (Table 3). It is reasonable to suppose that the page about "Macmillan," which is a publishing company, is more related to literature than the page about nations.

Table 2. ResourceRank values within pages.

RDF Resource	ResourceRank Score
"United State"	1.272
"Macmillan"	1.118
"Publishing company"	0.429

Table 3. Summary of ResourceRank used to compute WSPR.

Page	RDF Resource (ResourceRank Score)	WSPR Score
Macmillan	"Publishing company" (0.429) "Macmillan" (1.118)	1.547
United States	"United State" (1.272)	1.272

In summary, for the problem of importance value distribution, WSPR performed better than the other methods when the pages contained semantic information. WSPR obtains semantic resources from pages in a semantic-link-based Web structure and calculates resource rank values using the link weights among resources. Thus, WSPR is able to evaluate the rank values of Web pages based on the importance of semantic resources. The experiment showed the effectiveness of the ranking method using semantic metadata.

6. CONCLUSIONS

In this paper, we presented WSPR, a ranking method to improve the evaluation of link weights. WSPR uses semantic metadata in Web pages and adjusts the rank value propagation based on the evaluation of semantic links to obtain a more accurate ranking result. To utilize the semantic information, we transformed a hyperlink-based Web structure into a semantic-link-based Web structure. Our method effectively calculated the weight of the links to semantic resources and evaluated the importance of pages based on how many important resources they contained. Thus, in our method, pages that contained important information related to a given query could be

properly ranked highly, and the probability of providing highly scored meaningless pages was lowered. The comparative evaluation of WSPR against established baseline methods clearly demonstrated the acceptable performance of our method. Future directions to explore include using suitable methods to extract semantic information based on topic models, and building an automatic semantic metadata annotation system for Web pages not containing semantic metadata.

ACKNOWLEDGEMENT

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2014R1A1A1002236).

REFERENCES

[1] S. Brin and L. Page, "Reprint of: The Anatomy of a Large-Scale Hypertextual Web Search Engine," *Comput. Networks*, vol. 56, no. 18, pp. 3825–3833, 2012.

[2] J. M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment," *J. ACM*, vol. 46, no. 5, pp. 604–632, 1999.

[3] M. Kobayashi and K. Takeda, "Information Retrieval on the Web," *ACM Comput. Surv.*, vol. 32, no. 2, pp. 144–173, 2000.

[4] V. N. Gudivada, V. V. Raghavan, W. I. Grosky, and R. Kasanagottu, "Information Retrieval on the World Wide Web," *IEEE Internet Comput.*, vol. 1, no. 5, pp. 58–68, 1997.

[5] A. Singhal, "Modern Information Retrieval: A Brief Overview," *Bull. IEEE Comput. Soc. Tech. Comm. Data Eng.*, vol. 24, no. 4, pp. 35–43, 2001.

[6] K. Stein and C. Hess, "Information Retrieval in Trust-Enhanced Document Networks," *Semant. Web Min.*, pp. 65–81, 2006.

[7] W. Xing and A. Ghorbani, "Weighted PageRank Algorithm," *Proc. Second Annu. Conf. Commun. Networks Serv. Res. CNSR 2004*, pp. 305–314, 2004.

[8] P. Sharma, D. Tyagi, and P. Bhadana, "Weighted Page Content Rank for Ordering Web Search Result," *Int. J. Eng. Sci. Technol.*, vol. 2, no. 12, pp. 7301–7310, 2010.

[9] T. H. Haveliwala, "Topic-Sensitive PageRank," *Proc. 11th Int. Conf. World Wide Web*, pp. 517–526, 2002.

[10] G. Jeh and J. Widom, "Scaling Personalized Web Search," *Proc. twelfth Int. Conf. World Wide Web WWW 03*, pp. 271–279, 2003.

[11] RDF Working Group, "Resource Description Framework." [Online]. Available: http://www.w3.org/RDF/. [Accessed: 30-Jul-2015].

[12] J. Dean and S. Ghemawat, "MapReduce: Simplied Data Processing on Large Clusters," *Proc. 6th USENIX Symp. Opearting Syst. Des. Implement.*, pp. 137–150, 2004.

[13] H. Jun, W. Lee, D. Im, S. Lee, and H. Kim, "Weighted Semantic PageRank Using RDF Metadata on Hadoop," *Proc. Int. Conf. Internet Comput. Steer. Comm. World Congr. Comput. Sci. Comput. Eng. Appl. Comput.*, 2014.

[14] R. J. Bayardo, Y. Ma, and R. Srikant, "Scaling up all Pairs Similarity Search," *Proc. 16th Int. Conf. World Wide Web*, pp. 131–140, 2007.

[15] G. Salton and C. Buckley, "Term-weighting Approaches in Automatic Text Retrieval," *Inf. Process. Manag.*, vol. 24, no. 5, pp. 513–523, 1988.

[16] W3C Working Group, "RDFa Core 1.1 - Third Edition." [Online]. Available: http://www.w3.org/TR/rdfa-syntax/. [Accessed: 30-Jul-2015].

[17] R. Khare, "Microformats: The Next (Small) Thing on the Semantic Web?," *IEEE Internet Comput.*, vol. 10, no. 1, pp. 68–75, 2006.

[18] W3C Working Group, "HTML Microdata." [Online]. Available: http://www.w3.org/TR/2011/WD-microdata-20110405/. [Accessed: 30-Jul-2015].

[19] T. Steiner and M. Hausenblas, "How Google is Using Linked Data Today and Vision For Tomorrow," *Linked Data Futur. Internet Futur. Internet Assem. (FIA 2010), Ghent, Belgium*, 2010.

[20] "The Open Graph Protocol." [Online]. Available: http://ogp.me. [Accessed: 30-Jul-2015].
[21] S. Corlosquet, R. Cyganiak, A. Polleres, and S. Decker, "RDFa in Drupal: Bringing Cheese to the Web of Data," *Proc. 5th Work. Scripting Dev. Semant. Web ESWC*, 2009.
[22] R. De Virgilio, F. Frasincar, W. Hop, and S. Lachner, "A Reverse Engineering Approach for Automatic Annotation of Web Pages," *Multimed. Tools Appl.*, vol. 64, no. 1, pp. 119–140, 2013.
[23] M. Duma, "RDFa Editor for Ontological Annotation," *Proc. Student Res. Work. Assoc. with RANLP*, pp. 54–59, 2011.
[24] M. Samwald, E. Lim, P. Masiar, L. Marenco, H. Chen, T. Morse, P. Mutalik, G. Shepherd, P. Miller, and K. H. Cheung, "Entrez Neuron RDFa: A Pragmatic Semantic Web Application for Data Integration in Neuroscience Research," *Proc. Int. Conf. Eur. Fed. Med. Informatics*, pp. 317–321, 2009.
[25] A. Khalili, S. Auer, and D. Hladky, "The RDFa Content Editor - From WYSIWYG to WYSIWYM," *Proc. IEEE Signat. Conf. Comput. Software, Appl. COMPSAC*, pp. 531–540, 2012.
[26] S. Tramp, N. Heino, S. Auer, and P. Frischmuth, "RDFauthor: Employing RDFa for Collaborative Knowledge Engineering," *Proc. Cimiano, P., Pinto, H.S. EKAW 2010. LNCS*, vol. 6317, pp. 90–104, 2010.
[27] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann, "DBpedia - A Crystallization Point for the Web of Data," *J. Web Semant. Sci. Serv. Agents World Wide Web*, vol. 7, no. 3, pp. 154–165, 2009.
[28] N. Toupikov, J. Umbrich, R. Delbru, M. Hausenblas, and G. Tummarello, "DING! Dataset RankING Using Formal Descriptions," *Proc. WWW 2009 Work. Linked Data Web (LDOW 2009), Madrid, Spain*, 2009.
[29] "Hadoop." [Online]. Available: http://hadoop.apache.org. [Accessed: 30-Jul-2015].
[30] "Wikipedia." [Online]. Available: http://en.wikipedia.org/. [Accessed: 30-Jul-2015].
[31] K. Järvelin and J. Kekäläinen, "Cumulated Gain-based Evaluation of IR Techniques," *ACM Trans. Inf. Syst.*, vol. 20, no. 4, pp. 422–446, 2002.