An Efficient Manifold Ranking Approach for Monolithic Graphs and Semantic Networks

Prachya Boonkwan Dhanon Leenoi Thepchai Supnithi Akkharawoot Takhom
112 Phahonyothin Road, Khlong Nueng
Khlong Luang District, Pathumthani 12120, Thailand
Email: {prachya.boonkwan, dhanon.leenoi, thepchai.supnithi, akkharawoot.takhom}@nectec.or.th

***Abstract*—In recent decades, ranking has played a central role in large-scale information retrieval, preference analysis for recommendation systems, and influence analysis on social media. State-of-the-art ranking algorithms, e.g. `PageRank`, `HITS`, and their variants, traverse through a large network of connected items and compute their importance (a.k.a. *centrality*) by considering the incoming and outgoing links of each node. However, since these techniques take into account the centrality as an scalar quantity, it limits our analytical perspective to only one dimension — rank number. In this paper, we introduce an efficient multi-dimensional ranking approach that offers both node ranking and cluster analysis on any graph-based structure. We demonstrate that our approach is not only compatible with PageRank's scalar centrality but also manifests multi-dimensional spatial distribution of nodes. Therefore, it offers the possibility to perform cluster analysis on graph-based structures and keyword extraction via manifold centrality.**

***Index Terms*—manifold ranking • semantic network • PageRank Algorithm • eigenproblem • cluster analysis**

I. Introduction

Ranking is a non-trivial task in information retrieval and natural language processing. In large-scale information retrieval, search results are sorted by their relevance before returning to the user. Many recommendation systems sort search results by user preferences derived from a large network of items, users, and purchase history [1], [2]. In social media analysis, user impact and influence are analyzed to identify micro-influencers [3], [4]. In natural language processing, keyword extraction and automatic summarization can be done by sorting possible keywords by their significance [5].

State-of-the-art ranking algorithms, e.g. PageRank [6] and its variants such as SimRank [2], and HITS [7] traverse through a large network of connected items and compute their importance (a.k.a. *centrality*) by considering the incoming and outgoing links of each node. Especially in PageRank, ranking is considered an eigenproblem, where the network is converted into an adjacency matrix and the centrality of each node is efficiently computed by finding the dominant eigenvector of such matrix in an iterative fashion. The algorithm is proven to be stable and always converge. [8]

Despite its appealing stability and guarantee to converge, PageRank and its variants have three drawbacks. First, the computed centrality score is a scalar quantity; i.e. it has only one dimension. This limits our analytical perspective to only rank numbers and renders us unable to perform cluster analysis of nodes. Second, the method takes into account only one dominant eigenvector of the adjacency matrix, it discards all remaining non-dominant eigenvectors and the information contained. Third and last, the method assumes all links are of identical categories. This is not always the case in other graph-based structures such as semantic network whose links are categorically discriminated.

In this paper, we propose `ManifoldRank`, an efficient multi-dimensional ranking approach that offers both node ranking and cluster analysis on any graph-based structure. Henceforth, the notion of *manifold ranking* denotes multi-dimensional ranking of items in an n-dimensional Euclidean space. We believe that if we can compute a spatial distribution of nodes from their centrality, it will unleash the possibility to perform such task on any graph-based structure. This can be made possible by incorporating either non-dominant eigenvectors or dominant eigenvectors of the adjacency matrices of each relation type.

The rest of the paper is organized as follows. Section II provides some background on directed graph and PageRank Algorithm. Section III describes our manifold ranking approach in which multiple eigenvectors are incorporated into ranking. Section IV demonstrates that our approach yields results compatible with those of PageRank and also offers a spatial distribution of nodes according to their manifold centrality. We further discuss the experiment results in Section V. Finally, Section VI concludes the paper.

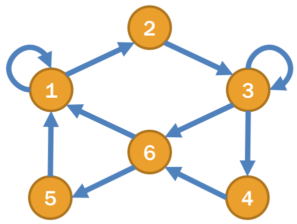
II. Background

A. Directed Graph and Adjacency Matrix

Let $G = (V, E)$ be a directed graph (as shown in Fig. 1(a)), where V is the set of all nodes (vertices) and E is the set of all edges linking from edge to edge. Any directed graph can be represented as an adjacency matrix $\mathbf{A} = [a_{ij}]$ (as shown in Fig. 1(b)), where each element a_{ij} represents the existence of an edge from node i to node j; i.e. $a_{ij} = 1$ if there exists such link in the graph, otherwise $a_{ij} = 0$.

B. PageRank Algorithm

PageRank [6] is a well-known iterative algorithm that computes the centrality of each node in any directed graph via random walk. In a nutshell, random walk (walking from a random starting point to all immediate neighbors, or pausing



(a) An example directed graph

target

$$\text{source}\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

(b) Its adjacency matrix

Figure 1. Directed graph

target

$$\text{source}\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 0 \end{bmatrix}$$

Figure 2. PageRank's adjacency matrix

at that point) on the graph reflects the count of visits of each node. The more steps the random walk takes, the more the importance of each node is intensified by the accumulated visit counts. If $\mathbf{A}$ is an adjacency matrix of dimension $N \times N$, a random walk of k steps is computed by

$$\mathbf{x}^{(k)} = (\mathbf{A} + \mathbf{I})^k \mathbf{x}^{(0)} \quad (1)$$

where $\mathbf{x}^{(0)}$ is a vector of dimension N denoting a starting point. The identity matrix $\mathbf{I}$ is added to $\mathbf{A}$ to represent a pause of random walk on each node.

In PageRank Algorithm, each edge of the graph is weighted in proportion to the specificity of the source node. The less outgoing edges it has, the more specific it is; i.e. we assign each element of the adjacency matrix $\mathbf{A}$ with

$$a_{ij} = 1/L_i \quad (2)$$

where L_i denotes the number of outgoing links from node i. For example, the directed graph in Fig. 1(a) will have the adjacency matrix in Fig. 2.

Once we obtain the adjacency matrix, we then compute the centrality of each node with Algo. 1. We start from a random point denoted by a normalized random vector $\mathbf{x}^{(0)}$. Here we set the probability of pausing at a node with a constant $\delta \in [0, 1]$ called a damping factor. We iteratively compute the random walk in step 6 and normalize the resultant vector in step 7. When the vector $\mathbf{x}^{(k)}$ converges within a small bound ϵ or the limit of iterations K has been reached, the algorithm terminates and returns $\mathbf{x}^{(k)}$.[1]

Each element of the returned vector $x_i^{(k)}$ is the centrality score of node i, which is proportionate to its accumulated

[1]In this paper, we set $\delta = 0.85$, $\epsilon = 10^{-4}$, and $K = 20$. We normalize the vectors with 2-norm, i.e. $\|\mathbf{x}\|_2 = \sqrt{\sum_{k=1}^{N} (x_k)^2}$. We compute the distance of two vectors $\mathbf{x}$ and $\mathbf{y}$ with distance$(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$.

Algorithm 1 `pagerank`$(\mathbf{A}, \delta, \epsilon)$

1: Let $\mathbf{x}^{(0)}$ be a random vector
2: $\mathbf{x}^{(0)} \leftarrow \mathbf{x}^{(0)} / \left\|\mathbf{x}^{(0)}\right\|_2$
3: Let $k \leftarrow 0$
4: **repeat**
5: $\quad k \leftarrow k + 1$
6: $\quad$ Compute $\mathbf{x}^{(k)} \leftarrow (1 - \delta)\mathbf{1} + \delta \mathbf{A}\mathbf{x}^{(k-1)}$.
7: $\quad \mathbf{x}^{(k)} \leftarrow \mathbf{x}^{(k)} / \left\|\mathbf{x}^{(k)}\right\|_2$
8: **until** distance$(\mathbf{x}^{(k)}, \mathbf{x}^{(k-1)}) < \epsilon$ **or** $k \geq K$
9: **return** $\mathbf{x}^{(k)}$

visit count, and this score is then used for ranking. Since the iteration is equivalent to performing the Power Method [9], the resultant vector $\mathbf{x}^{(k)}$ is thus the dominant eigenvector of the adjacency matrix. Note that each element of any eigenvector may be negative.

C. Spectral Decomposition

An *eigenvector* of a linear transformation is a non-zero vector that changes only by a scalar factor when that linear transformation is applied to it. In the N-dimensional vector space, the linear transformation becomes a square matrix $\mathbf{A}$ of dimension $N \times N$, and the vector is a column vector $\mathbf{x}$ of dimension N, such that

$$\mathbf{A}\mathbf{x} = \lambda \mathbf{x} \quad (3)$$

where λ is a non-zero scaling factor called the *eigenvalue* associated with the eigenvector $\mathbf{v}$. Any $N \times N$ matrix always has N pairs of eigenvectors and associated eigenvalues. The set of these pairs is called the *eigensystem* of a linear transformation. The pair $(\mathbf{x}, \lambda)$ whose eigenvalue λ is the largest is called the dominant eigenvector and eigenvalue, respectively. Assuming all eigenvectors $\mathbf{x}$ are unit vectors, the *spectral decomposition* of $\mathbf{A}$ is the expansion in terms of eigenvectors and eigenvalues:

$$\mathbf{A} = \sum_{i=1}^{N} \lambda_i \mathbf{x}_i \mathbf{x}_i^\top \quad (4)$$

where each eigenvalue λ_i corresponds to $\mathbf{x}_i$, and all eigenvalues are sorted: $\lambda_1 > \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_N > 0$.

III. Manifold Ranking

A. Motivation

Our approach is generally motivated by the spectral decomposition. It is supported by Kleinberg's [7] suggestion on considering multiple eigenvectors as a way to obtain authorities within multiple node communities. In Eq 4, any adjacency matrix, which is a square matrix, is composed of N pairs of eigenvectors and eigenvalues. Discarding some non-dominant eigenvectors and eigenvalues will result in approximating the adjacency matrix, causing information loss in the process. Only considering the dominant eigenvector, PageRank jettisons all centrality scores in the remaining eigenbases that can discriminate items in the same rank in finer details.

We are motivated to integrate the traditional PageRank Algorithm with non-dominant eigenvectors and eigenvalues, resulting in the notion of *manifold ranking*, denoting multi-dimensional ranking of items in an n-dimensional Euclidean space. Its advantages have two folds. First, our centrality ranking is still compatible with the original PageRank. Second, it also manifests a multi-dimensional spatial distribution of nodes in the graph.

B. Monolithic Manifold Ranking

Let's employ some non-dominant eigenvectors in addition to the dominant eigenvector. Since each eigenvector determines node centrality scores on its eigenbasis, we imply that considering these scores on multiple eigenbases may yield finer-grained ranking. Moreover, we will weight the scores on each eigenbasis with its own eigenvalue.

Assume that an input directed graph G is monolithic; i.e. all edges are not discriminated by relation types. We extract an adjacency matrix $\mathbf{A}$ out of G, and follow Algo. 2. We decompose the adjacency matrix into m dominant eigenvectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ and the corresponding eigenvalues $\lambda_1 > \lambda_2 \geq \ldots \geq \lambda_m > 0$. In step 4, we compute the eigenvalue λ_k of each $\mathbf{x}^{(k)}$ with the Rayleigh quotient. In step 5, we anihilate the impact of previously found eigenvectors with Hotelling's deflation method. This algorithm yields a rank matrix

$$\mathbf{R} = \frac{1}{Z}\begin{bmatrix} \lambda_1 \mathbf{x}^{(1)\top} \\ \vdots \\ \lambda_m \mathbf{x}^{(m)\top} \end{bmatrix} \tag{5}$$

where each column vector $\mathbf{r}_i$ of $\mathbf{R}$ is the rank vector of each i-th node, and $Z = \sum_{k=1}^{m} \lambda_k$ is the normalizing factor. Thus, for any i-th node, the centrality score on the j-th eigenbasis is $r_{ij} = \lambda_j x_i^{(j)}/Z$.

Algorithm 2 $\texttt{manifoldrank\_monolithic}(\mathbf{A}, m, \delta, \epsilon)$

1: Let $\mathbf{A}^{(1)} = \mathbf{A}$
2: **for** $k = 1$ **to** m **do**
3: Compute $\mathbf{x}^{(k)} \leftarrow \texttt{pagerank}(\mathbf{A}^{(k)}, \delta, \epsilon)$
4: Compute $\lambda_k \leftarrow \frac{\mathbf{x}^{(k)\top}\mathbf{A}^{(k)}\mathbf{x}^{(k)}}{\mathbf{x}^{(k)\top}\mathbf{x}^{(k)}}$
5: $\mathbf{A}^{(k+1)} \leftarrow \mathbf{A}^{(k)} - \lambda_k \mathbf{x}^{(k)}\mathbf{x}^{(k)\top}$
6: **end for**
7: Let $\mathbf{R} = \frac{1}{Z}\begin{bmatrix} \lambda_1 \mathbf{x}^{(1)\top} \\ \vdots \\ \lambda_m \mathbf{x}^{(m)\top} \end{bmatrix}$, where $Z = \sum_{k=1}^{m} \lambda_k$
8: **return** $\mathbf{R}$

Note that each node now has a rank vector rather than a scalar centrality score. For ranking purposes, we can then compute a scalar score out of a rank vector $\mathbf{r}_i$ by calculating its size with p-norm:

$$\|\mathbf{r}_i\|_p = \left[\sum_{j=1}^{m} |r_{ij}|^p\right]^{1/p} \tag{6}$$

where $p \geq 1$.[2] Note the absolute sign $|\cdot|$; therefore, we consider only the absolute value of each element r_{ij}. Furthermore, the spatial distribution of these rank vectors allows us to perform cluster analysis with respect to node centrality. We will see that nodes with similar PageRank scores tend to constellate in the same cluster in Section IV.

C. Semantic Manifold Ranking

In the case where an input directed graph is a semantic network, it is compulsory to discriminate its edges with relation types. However, this need has never been addressed in the traditional PageRank Algorithm. We alleviate this issue by segregating edges with their relation types and ranking each group separately. Since the rankings of each group are on different eigenbases, we weight them with their eigenvalues.

Given an input semantic network G, we extract an adjacency matrix for each relation type, resulting in a list of n adjacency matrices $[\mathbf{A}_n]$. Following Algo. 3, we decompose each $\mathbf{A}_k$ into the dominant eigenvector $\mathbf{x}^{(k)}$ for each relation type k. Then we compute the corresponding eigenvalue $\lambda_k > 0$ with the Rayleigh quotient. This algorithm also yields a rank matrix $\mathbf{R}$, which is similarly computed by Eq 5.

Algorithm 3 $\texttt{manifoldrank\_semantic}([\mathbf{A}_n], \delta, \epsilon)$

for each relation type $k = 1$ **to** n **do**
Computer $\mathbf{x}^{(k)} \leftarrow \texttt{pagerank}(\mathbf{A}_k, \delta, \epsilon)$
Compute $\lambda_k \leftarrow \frac{\mathbf{x}^{(k)\top}\mathbf{A}_k\mathbf{x}^{(k)}}{\mathbf{x}^{(k)\top}\mathbf{x}^{(k)}}$
end for
Let $\mathbf{R} = \frac{1}{Z}\begin{bmatrix} \lambda_1 \mathbf{x}^{(1)\top} \\ \vdots \\ \lambda_m \mathbf{x}^{(m)\top} \end{bmatrix}$, where $Z = \sum_{k=1}^{m} \lambda_k$
return $\mathbf{R}$

Note that each node has a rank vector of dimension n. This vector collects the weighted centrality scores for each relation type. For ranking purposes, we can compute a scalar score out of a rank vector $\mathbf{r}_i$ by p-norm in Eq 6. The spatial distribution of these rank vectors also allows us to perform cluster analysis, as we will see in Section IV.

IV. Experiments

A. Settings

Datasets: We use four semantic networks of Linked Open Data (shown in Table I) to evaluate the manifold ranking. All of these datasets are RDF graphs that contain a set of triples (s, v, o), where s is a subject, v a verb or a relation, and o an object, all of which encoded as URIs (uniform resource identifiers). The URIs of distinct subjects and objects will become nodes in the graph, and all edges are categorically discriminated by the URIs of distinct verbs. In this paper, we classify semantic networks into two categories: small (containing up to 20,000 nodes) and large (containing more than 20,000 nodes). We deliberately vary the number of nodes to investigate the effects of this factor.

[2] We set $p = 2$ in this paper.

Table I
DATASET STATISTICS

	Descriptions	URIs	Rels	Triples
JTexts	Bibliographical LOD of textbooks used in elementary & secondary education in Japan [10]	14,961	12	51,807
Nobel	LOD about every Nobel prize since 1901 [11]	19,279	21	52,222
Nomisma	LOD of numismatic concepts linked to other resources [12]	41,586	28	77,068
WN30	WordNet 3.0: LOD of English lexicons grouped as synonym sets, a.k.a. synsets, and relations among these synsets [13]	117,663	8	240,744

Table II
SIMILARITY (r_s) BETWEEN PAGERANK AND MANIFOLD RANKING

	Monolithic	Semantic
JTexts	0.9550	**0.9967**
Nobel	0.9820	**0.9962**
Nomisma	**0.9948**	0.9830
WN30	**0.9663**	0.9281

Parameter setup: We set the damping factor $\delta = 0.85$, the acceptable error rate $\epsilon = 10^{-4}$, and the limit of iterations $K = 20$ in all experiments. In semantic manifold ranking, we extract only one eigenvector for each relation type; therefore, the size of the rank vectors equals to the number of relation types. We compute the scalar centrality score out of the rank vectors in Eq 6 using 2-norm; i.e. $p = 2$.

Evaluation metrics: It is necessary to demonstrate how similar our ranking approach is to the PageRank Algorithms. We quantitatively measure the similarity of both ranking systems with Spearman's rank correlation coefficient [14]:

$$r_s = \frac{\operatorname{cov}(\mathrm{rg}_{\mathbf{x}}, \mathrm{rg}_{\mathbf{y}})}{\sigma_{\mathrm{rg}_{\mathbf{x}}} \sigma_{\mathrm{rg}_{\mathbf{y}}}} \tag{7}$$

where $\mathrm{rg}_{\mathbf{x}}$ and $\mathrm{rg}_{\mathbf{y}}$ are ranking orders of two raw scores x_i and y_i of a sample of size N. $\operatorname{cov}(\mathrm{rg}_{\mathbf{x}}, \mathrm{rg}_{\mathbf{y}})$ is the covariance of the rank variables

$$\operatorname{cov}(\mathrm{rg}_{\mathbf{x}}, \mathrm{rg}_{\mathbf{y}}) = \frac{1}{N} \sum_{i=1}^{N} \left[(\mathrm{rg}_{\mathbf{x}}^{(i)} - \bar{r}_{\mathbf{x}})(\mathrm{rg}_{\mathbf{y}}^{(i)} - \bar{r}_{\mathbf{y}}) \right] \tag{8}$$

where $\bar{r}_{\mathbf{x}}$ and $\bar{r}_{\mathbf{y}}$ are the average ranks of all x_i and all y_i, respectively. $\sigma_{\mathrm{rg}_{\mathbf{x}}}$ and $\sigma_{\mathrm{rg}_{\mathbf{y}}}$ are standard deviations of $\mathrm{rg}_{\mathbf{x}}$ and $\mathrm{rg}_{\mathbf{y}}$, respectively. The more r_s it is, the more similar the ranking orders are.

B. Experiment 1: Ranking

We compared our method against the traditional PageRank via the Spearman's rank correlation coefficient r_s. For each semantic network, the similarity of ranking orders generated from the two methods of manifold ranking, i.e. monolithic and semantic, is measured against the ranking order generated from PageRank.

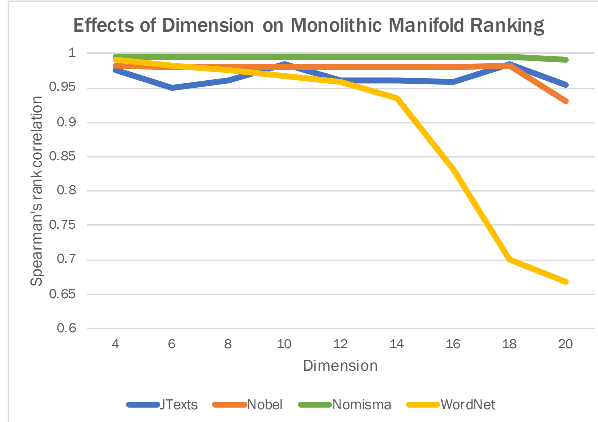


Figure 3. Effects of dimensions m on the monolithic manifold ranking

The results in Table II suggest that the number of nodes seems to affect the similarity of the resultant rank ordering. Here we chose to extract 10 eigenvectors in monolithic manifold ranking, i.e. $m = 10$. The semantic method is more similar than the monolithic method in small semantic networks, while the first is slightly less similar than the latter in large semantic networks. When applying our approach on a very large semantic network e.g. WN30, the similarity scores are relatively inferior to those of smaller networks. We suspect that our scalar ranking score computed by Eq 6 may be vunerable to negative centrality scores.

We further investigated the effects of m (the number of incorporated eigenvectors) in monolithic manifold ranking. In Fig. 3, we varied m from 4 to 20 and measured the similarity to PageRank. The similarity slightly changes in the case of JTexts, Nobel, and Nomisma. However, the similarity for WN30 plummets after $m = 10$. We suspect that incorporating non-dominant eigenvectors to ranking may have increased the susceptibility to negative centrality scores, causing a similarity plunge.

C. Experiment 2: Cluster Analysis

We studied the spatial distribution of rank vectors in tandem with PageRank's scalar centrality scores. In Fig. 4, we performed spectral cluster analysis of rank vectors obtained in Experiment 1 with t-SNE [15]. This technique spectrally projects a set of input vectors into a lower-dimensional space, while preserving all constellations as in the higher-dimensional space. Each dot denotes a projection of a rank vector onto the 2D space. For each dot, a PageRank centrality score can be determined by the dot's color: violet (lowest) < blue < teal < green < yellow < orange < red (highest). The big dots represent the top-20 list of ranking.

From the scatter plots, we found that rank vectors produced from the monolithic manifold ranking seem more dispersed than those produced from the semantic manifold ranking, e.g. Fig. 4(e) vs. Fig. 4(f). The dispersion is even more profound across the board when we investigated the top-20 nodes (big dots), e.g. Fig 4(a) vs. Fig 4(b). These big dots tend to constellate more tightly in semantic manifold ranking, because central nodes usaly have high centrality scores in every relation type.

We observed that low-ranked vectors (in violet) seem to separate from the higher-ranked ones quite clearly in both ranking methods, especially in Figs. 4(c), 4(d), and 4(f). That is also because non-central nodes usually have low centrality scores in most relation types, resulting in clusters of nodes of the same ranks. Nevertheless, higher-ranked nodes seem to intermingle with other ones of different ranks in the monolithic method, particularly in Figs. 4(e) and 4(g). This is due to spikes in non-dominant eigenvectors that perturb the spatial distribution of nodes according to their rank vectors.

Interestingly, for a very large semantic network with few relation types, e.g. WN30, we found that the monolithic method has a better spatial distribution of rank vectors than the semantic method does (Fig. 4(g) vs. Fig. 4(h)). This is because the size of rank vectors in the monolithic method ($m = 10$) is larger than that in the semantic method ($n = 8$), offering finer-grained ranking and more distinct separation.

V. Discussion

In this section, we will discuss three findings derived from the experiment results: the behaviors of rank vectors, the size of rank vectors, and the scalar rank score.

Behaviors of rank vectors: We can generally classify rank vectors into four types. Type 1 (*the top-notch*) has high centrality scores in most or all dimensions. Type 2 (*the spiky*) generally has middle-to-low centrality scores in the majority of dimensions and have high scores (or spikes) in the remaining dimensions. Type 3 (*the common*) has middle-to-low centrality scores in almost all dimensions. Finally, Type 4 (*the low-ranking*) has very low centrality scores in almost all dimensions. Among these types, the rank vectors of Group 3 are the most problematic for cluster analysis, because they tend to mingle with each other and form a mistakenly mixed group. Weighting each dimension with its corresponding eigenvalue as in Eq 5 certainly helps mitigate this issue in some degree. However, weight assigning for each non-dominant eigenvector is still an open question and may be task-specific.

Size of rank vectors: In monolithic manifold ranking, non-dominant eigenvectors convey additional information and make ranking finer-grained. Weighting them with their corresponding eigenvalues makes this additional information more subtle, not overpowering the PageRank's centrality scores. However, incorporating excessive eigenvectors may diminish the effects of these scores if non-dominant eigenvalues become large. Choosing the right number of non-dominant eigenvectors may be a bit tricky.

Scalar rank score: We found that employing p-norm as the scalar score in Eq 6 makes ranking susceptible to negative centrality scores. In our experiments, the similarity of ranking on small to medium-sized semantic networks are close to 1.0, while that of a very large semantic network are approximately 0.72. This is because there is higher chance to find negative centrality scores on a very large semantic network. To circumvent this issue, we may use the p-pseudonorm instead:

$$\text{pseudonorm}(\mathbf{r}_i, p) \quad = \quad \|\text{sigmoid}(\mathbf{r}_i)\|_p \tag{9}$$

where $p \geq 1$ is a positive number. However, computing a scalar rank score out of a rank vector is still a tough challenge.

VI. Conclusion

We proposed an efficient manifold ranking approach that offers both PageRank-styled ranking and cluster analysis on any graph-based structures. The experiment results showed that our approach is compatible with PageRank's scalar centrality and manifests multi-dimensional spatial distribution of nodes. Our method offers the possibility to perform cluster analysis and keyword extraction via manifold ranking.

Our future work remains as follows. First, we will improve the process of eigenvector weighting by taking into account task-specific information, e.g. knowledge-graph completion. Second, we will explore the effects of incorporating more non-dominant eigenvectors in monolithic manifold ranking. Third, we will consider various types of norm, as well as the later proposed p-pseudonorm, as a way to tackle negative centrality scores. Fourth and finally, we will endeavor to improve the task of network community detection with manifold ranking.

References

[1] A. Passant, "Measuring semantic distance on linking data and using it for resources recommendations," in *AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*, 2010.

[2] P. T. Nguyen, P. Tomeo, T. D. Noia, and E. D. Sciascio, "An evaluation of SimRank and personalized PageRank to build a recommender system for the web of data," in *Proceedings of the International World Wide Web Conference Committee (IW3C2)*, pp. 1477–1482, 2015.

[3] L. Huang and Y. Xiong, "Evaluaton of microblog users' influence based on PageRank and users behavior analysis," *Advances in Internet of Things*, vol. 3, pp. 34–40, 2013.

[4] B. Xiang, Q. Liu, E. Chen, H. Xiong, Y. Zheng, and Y. Yang, "PageRank with priors: An influence propagation perspective," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pp. 2740–2746, 2013.

[5] R. Mihalcea and P. Tarau, "TextRank: Bringing order into texts," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2004.

[6] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the web," Tech. Rep. 422, Stanford University, 1999.

[7] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[8] M. Bianchini, M. Gori, and F. Scarselli, "Inside PageRank," *ACM Transactions on Internet Technology*, vol. 5, pp. 92–128, February 2005.

[9] R. von Mises and H. Pollaczek-Geiringer, "Praktische verfahren der gleichungsauflösung," *ZAMM - Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 9, pp. 152–164, 1929.

[10] "Japanese textbooks LOD – `https://w3id.org/jp-textbook`."

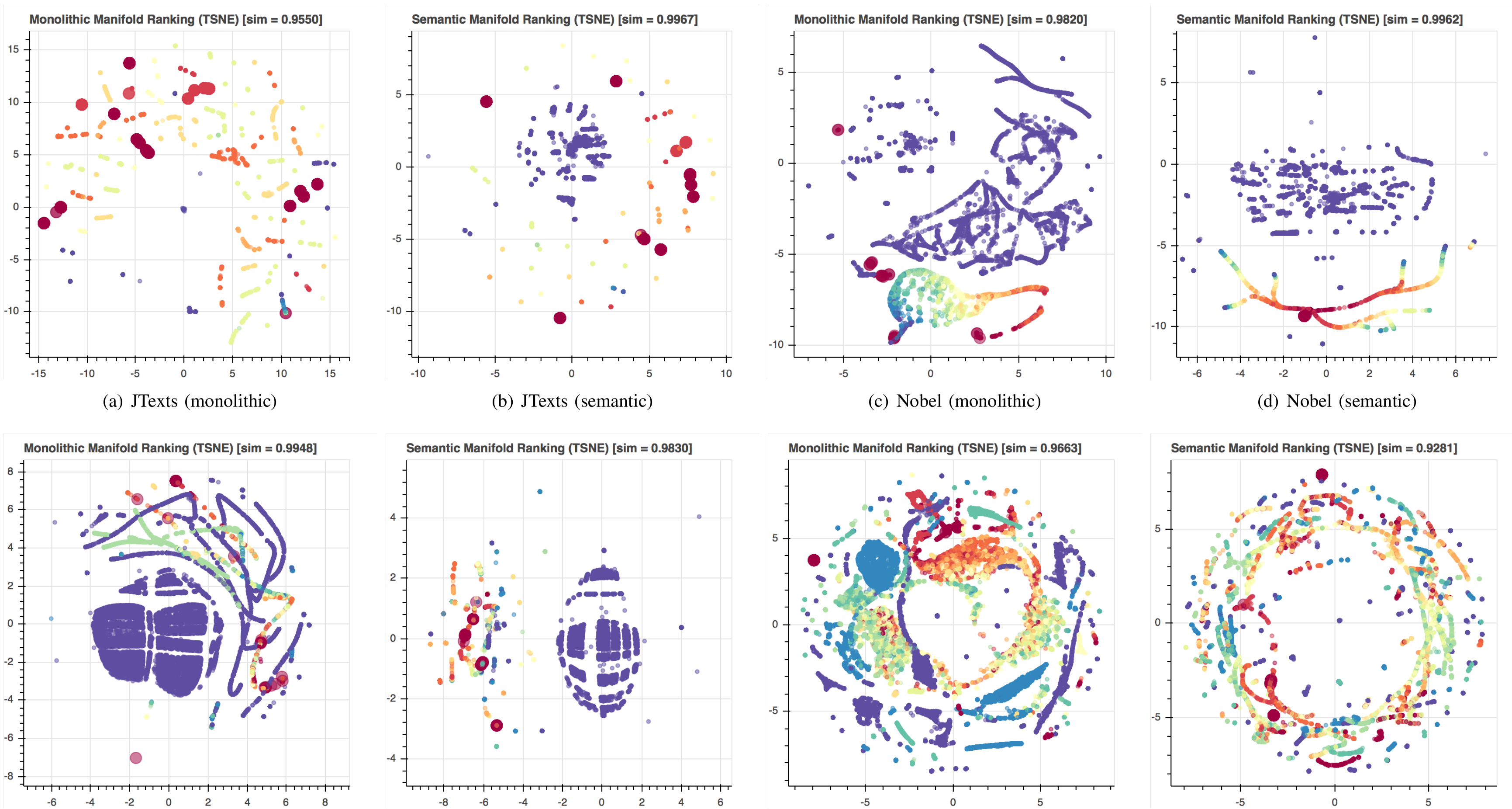
[11] "Nobel prize LOD – `http://www.nobelprize.org/nobel_organizations/nobelmedia/nobelprize_org/developer/manual-linkeddata/terms.html`."

[12] "Nomisma LOD Project – `http://nomisma.org`."

[13] "WordNet 3.0 – `http://semanticweb.cs.vu.nl/lod/wn30`."

[14] C. Spearman, "The proof and measurement of association between two things," *American Journal of Psychology*, vol. 15, pp. 72–101, 1904.

[15] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, November 2008.



(a) JTexts (monolithic) (b) JTexts (semantic) (c) Nobel (monolithic) (d) Nobel (semantic)

(e) Nomisma (monolithic) (f) Nomisma (semantic) (g) WN30 (monolithic) (h) WN30 (semantic)

Figure 4. Cluster analysis of rank vectors. Color values determine the ranking of PageRank scores: violet (lowest) < blue < teal < green < yellow < orange < red (highest). Big dots denote concepts in the top-20 ranks. The rank vectors are spectrally reduced to two dimensions using t-SNE. In monolithic manifold ranking, we extract only 10 eigenvectors; i.e. we set $m = 10$.