ÇOMÜ LJAR (2020) Cilt 1 Sayı 2 (61-70)
e-ISSN: 2717-8285
Geliş Tarihi: 06.10.2020
Kabul Tarihi: 02.12.2020
Araştırma Makalesi (Research Paper)

A Comparison of Graph Centrality Algorithms for Semantic Distance

Enis ARSLAN[1]* **Erhan TURAN[2]**
Çağatay Neftali TÜLÜ[3] **Umut ORHAN[1]**

[1] Çukurova Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 01380, Adana
[2] Osmaniye Korkut Ata Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 80000, Osmaniye
[3] Adana Alparslan Türkeş Bilim ve Teknoloji Üniversitesi, Bilgi İşlem Daire Başkanlığı, 01250, Adana

*Sorumlu yazar: enisarslan@gmail.com

Abstract

Semantic networks are kind of datasets used for natural language processing (NLP). Distance measurement for semantic networks, which are generally based on a graph structure, is a vital requirement for semantic analysis on concepts. Centrality measures can be used for calculating the semantic distance between concepts in a semantic network. In this paper, we evaluated graph centrality algorithms including PageRank, Hyperlink-induced Topic Search (HITS), and Betweenness Centrality on a semantic network, which was created from a Turkish dictionary dataset. Centrality measures special to these algorithms are used to calculate the semantic distance between synonym pairs in the semantic network. Also, we have used a simple centrality method beside the other three popular centrality algorithms to find out the most accurate and cost-effective method on our semantic network. Working on a bipartite model of the network which increases the complexity of implementation for centrality algorithms and performing calculations on a semantic network, that can be expanded with new nodes and edges, are two major challenges to overcome. Considering all these conditions, results from each algorithm are compared to pick out an optimal method for the semantic network.

Key Words: Betweenness Centrality, HITS, PageRank, Semantic Distance, Semantic Networks

Graf Merkezilik Algoritmalarının Anlamsal Mesafe için Karşılaştırılmaları

Özet

Anlamsal ağlar, doğal dil işleme (DDİ) için kullanılan graf tabanlı veri kümeleridir. Anlamsal ağlarda mesafe ölçümü ise, kavramların ağ içinde ilişkiler ile birbirine bağlılığının anlamsal analizi için çok önemli bir yere sahiptir. Bağlantılılık ölçümleriyle elde edilen değerler, anlamsal ağlardaki kavramlar arasındaki mesafe hesaplamaları için kullanılabilinir. Bu çalışmada, PageRank, Hyperlink-induced Topic Search (HITS) ve Arasındalık Merkeziliği graf bağlantılılık algoritmaları, Türkçe sözlükteki kavramlardan oluşturulan anlamsal ağ üzerinde uygulanmış ve elde edilen değerler ile anlamsal ağdaki eş anlamlı sözcükler arasındaki mesafe hesaplanmıştır. Bu üç önemli graf bağlantılılık algoritmaları, bu çalışmada kullanılan anlamsal ağ için tasarlanmış olan temel bir bağlantılılık yöntemiyle karşılaştırılmıştır. İki parçalı graf tasarımı ile oluşturulmuş olan Türkçe Sözlük anlamsal ağı üzerinde geleneksel graf bağlantılılık algoritmalarının uygulanması daha karmaşık hale gelmektedir. Uygulama esnasında gereken işleme zamanını arttırması, ayrıca ağa eklenecek olan yeni kavramlar ve bağlantılar sonrası ağın tekrar anlamsal mesafe için hesaplamalara ihtiyaç duyması, bağlantılılık algoritmalarının karşılaştığı iki önemli sorundur. Bu zorluklar ve anlamsal ağın iki parçalı graf yapısı göz önüne alındığında, her bir algoritma ile elde edilen sonuçlar karşılaştırılmış ve tasarlanan anlamsal ağ için en verimli yöntem bulunmaya çalışılmıştır.

Anahtar Kelimeler: Arasındalık Merkeziliği, HITS, PageRank, Anlamsal Mesafe, Anlamsal Ağlar

ORCID ID (Yazar sırasına göre/By author order)
0000-0002-2609-3925 0000-0002-2953-2698 0000-0002-4462-3707 0000-0003-1882-6567

1.Introduction

Semantic networks of words are frequently used in natural language processing (NLP) studies like machine translation, automatic summarization and sentiment analysis. These semantic studies use graphic data generated by words and semantic relationships between them. WordNet (Miller, 1995) is known as the most advanced semantic network among other semantic networks created for any natural language. and it is based on senses with their synonym sets.

Veronis and Ide (1990), designed an unweighted lemma-sense graph to create a semantic network to get better results for word sense disambiguation. Graph data with weighted relations between nodes may contain more patterns than unweighted graphs (Li et al., 2011). Thus, generating a semantic network with weighted relations is essential to properly analyze the distance between words. A semantic network designed with the model proposed in Veronis and Ide's study can be improved by weighted relations between lemmas and their senses.

Distance in an unweighted graph can be generated by computing the centrality of the nodes in the graph data and can be accomplished by using centrality algorithms for graphs. The main purpose of centrality algorithms is to detect the most important nodes in a graph, to find out the important nodes, these algorithms calculate weights on the nodes and relations of the graph. Most important nodes detected with centrality algorithms on a semantic network can be interpreted as generic terms of the word taxonomy of that semantic network. Greater centrality weights will point out the more general words, while lower weights will be defined for specific words of the semantic network.

In this study, a simple method designed for this semantic network and three well-known centrality algorithms, PageRank, Hyperlink-induced Topic Search (HITS), and Betweenness Centrality are evaluated to compute weights for words to score them from specific to generic terms. Contrary to the purpose of centrality algorithms, if a node is weighted with higher values it will be assumed as a generic word and less important in a semantic network for disambiguation of word senses. Networks designed in bi-partite graph architectures are considered computationally expensive when they are evaluated on popular graph connectivity algorithms like HITS, PageRank and Betweenness Centrality algorithms.

When networks designed in bi-partite graph architectures are considered many of the popular graph centrality algorithms like HITS, PageRank and Betweenness Centrality algorithms are computationally expensive. A graph centrality algorithm should be simple and neat to perform graph operations like "insert node" / "delete node" with low complexity and high efficiency.

In this study, we present an evaluation of the popular graph centrality algorithms, PageRank, HITS and Betweenness Centrality on a given semantic network for measuring the semantic distance between synonyms. We aimed to observe the behaviors of these algorithms on the bi-partite network structure and interpret the results in comparison with a modified Mention-Sense algorithm. This paper is organized as follows: In Section 2, material information and method are given. In Section 3, results and discussion of the results are provided. The paper ends with the conclusion of the study in Section 4.

2.Material and Method

2.1. Material

In our previous work, a semantic network was built from a Turkish dictionary (Turan and Orhan, 2018). "Türkçe Güncel Sözlük", which is the main actual dictionary of the Turkish Language Association (TLA) was used as a dataset to create a semantic network with the lemmas and their senses.

The semantic network structure consists of two primary nodes and various semantic relations between these nodes. The “Lemma” nodes represent the head of each item in the TLA dictionary on the network, while the “Sense” nodes represent each definition that exists in the definitions section of the lemma item. Since each item has at least one definition of its own, there is a semantic relation link labeled "SENSE" with at least one sense node of each “LEMMA” node on the semantic network. If there are homonyms of an item, the lemma nodes with the same name but with different homonym order are created. In this way, it is easily interpreted as different words in the analysis of sense ambiguity between the homonyms. As shown in Figure-1 below, two homonyms for the word kurt["worm"] (I) and kurt["wolf"] (II) lemma nodes and their sense nodes are independently associated.

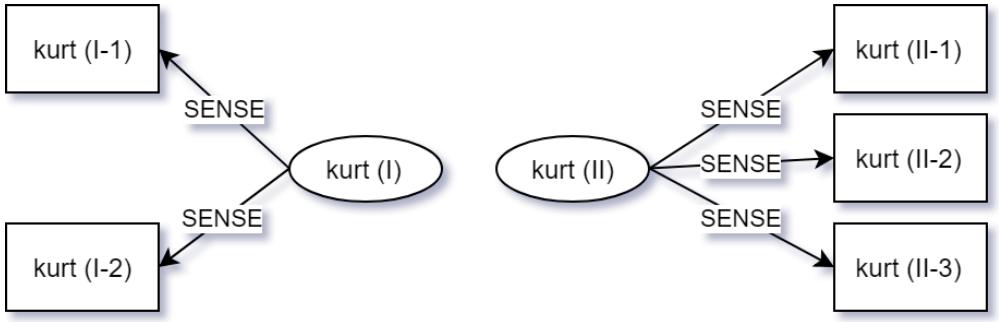

Figure 1. Two homonyms and their senses for word "kurt" on the semantic network

The next essential relation in the semantic network is the connection from sense nodes to the other lemma nodes. In a sense node, the definition content is used in the process of associating with other lemma nodes. Each word or phrase in a definition sentence is connected with that sense if the word or the phrase in the definition exists as a lemma node in the semantic network. This relationship is labeled with the name "MENTIONS". "SENSE" and "MENTIONS" are the basic relations used to establish a semantic network among all nodes in the graph structure. Figure 2 shows an example with lemma node “ev” ["home"], its sense nodes and other lemma nodes that are associated with the sense nodes.

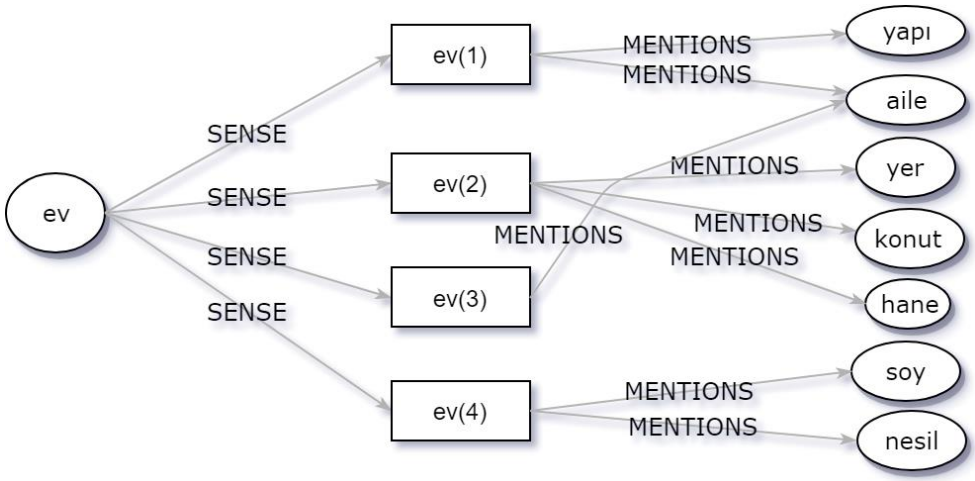

Figure 2. Lemma node ‘ev’ and its sense nodes connected to other lemmas with MENTIONS relations

These two basic relations “SENSE” and “MENTIONS”, structure a bipartite graph model that a “SENSE” relation connects a lemma node to a sense node while a “MENTIONS” node connects a sense node to a lemma node. And unfortunately, not all the lemma nodes have at least one “MENTIONS” relation connected, which prevents the semantic network to be a connected graph. Thus, the semantic network needs more relations to decrease the sparsity of the graph structure which can be accomplished by adding “DERIVED”, “COMPOUND” and “PHRASE” relations between the lemma

nodes. The “DERIVED” relation connects a lemma node to its derived lemma nodes with derivational affixes. And “COMPOUND” relation connects a lemma node to another lemma node which is a compound word that contains the source lemma node. “PHRASE” relation created between two lemma nodes which connect a lemma node to the other lemma node where both are part of the phrase. These relations connect lemma nodes, and they ignore the sense nodes in the semantic network. Besides these relations, “SYNONYM” relations are extracted from the definitions of the sense nodes and connected from a sense node to another sense node. Figure 3, shows all the relation types in the semantic network for the lemma node “ev”.

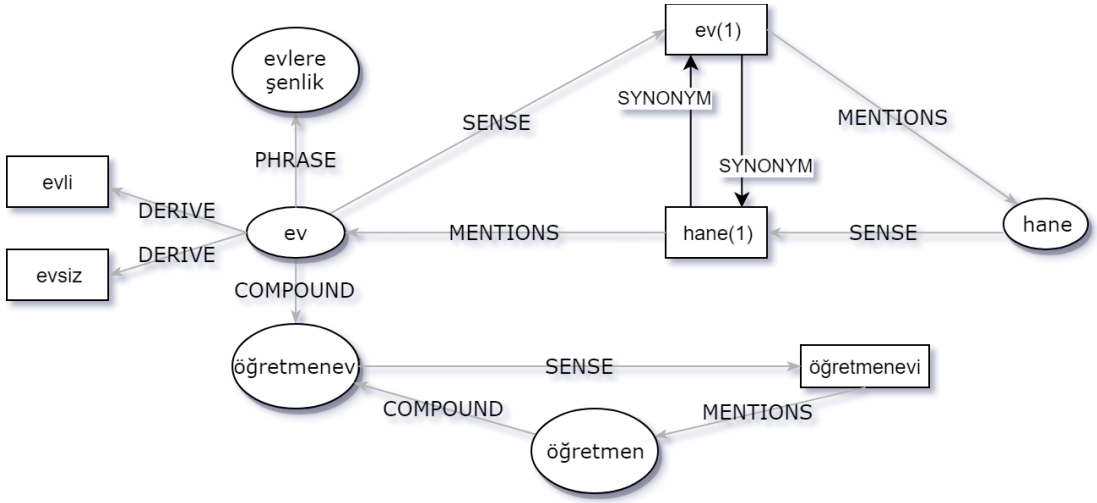

Figure 3. Lemma node ‘ev’ with an example of all type of relations

As mentioned before “SENSE”- “MENTIONS” relations create a bipartite graph model while other relations are established between lemma nodes. This leads to implementation problems for the PageRank, HITS and Betweenness Centrality algorithms. Two different node types are not suitable for general graph centrality algorithms to apply. To overcome this problem, the graph structure is changed by ignoring sense nodes and each lemma is directly connected to the lemma that is mentioned in its definitions for evaluation of the centrality algorithms. And also all relation types are merged into one relation named “RELATED” to create a simple one-typed node and one-typed relation graph structure.

2.2. Methods

In this study, three notable graph centrality algorithms and a method, proposed for the semantic network, are evaluated to find a suitable method for the calculation of the semantic distance between any lemma nodes in the network. Most widely used centrality algorithms (Newman, 2018), PageRank, HITS and Betweenness Centrality methods, are applied to the graph data with slightly modified versions.

2.2.1. PageRank

The PageRank algorithm is designed to be used as a search engine on the public network (Brin and Page, 1998). It works by calculating the number of links to and from a web page. PageRank algorithm is an essential centrality algorithm and beyond web search, its generally applicable mathematical base allows to apply to a graph of any domain (Gleich, 2015). PageRank algorithm is an iterative algorithm that requires very intensive processing power in large graphs, which led to versions of the algorithm working in parallel (Manaskasemsak and Rungsawang, 2005). The equation of the algorithm is shown below.

$$PR(A) = (1-d) + d\left(\frac{PR(T_1)}{C(T_1)} + \ldots + \frac{PR(T_n)}{C(T_n)}\right) \quad (1)$$

PR(A) is the PageRank value of *node A*, *d* is the damping constant value between 0 and 1, $T_{1..n}$ are the source nodes that connect to *node A* and $C(T_n)$ represents the number of outgoing links to other nodes for *node* T_n.

2.2.2. HITS

HITS algorithm (Kleinberg et al., 1999; Savić et al., 2019) is designed for web pages with links associated with each other over the Internet like PageRank. And also can be used for any relational network created with a single type of node. A modified version of the HITS algorithm that is adapted for a bipartite graph is used to evaluate web service reputation (Tibermacine et al., 2019). HITS has two weights for each node. These weights are defined as the Hub and the Authority values. Hub value represents the importance of a node in linking to other nodes, and the Authority value of a node defines total Hub values of nodes that link to that node. If the Hub value is high on a node, the nodes that it points will be valuable. If the Authority value is high on a node, the Hub values of the other nodes that point to the node are valuable. Designed for Internet pages, this algorithm aims to find out whether a page is valuable as a source of knowledge or as a gateway for knowledge. The calculations for Hub and Authority values for each node must be applied several times until the results reach a certain convergence.

$$a_j^{(0)} = 0, \quad h_j^{(0)} = 1 \quad (2a)$$

$$a_i^{(t+1)} = \sum_{j \to i} h_j^{(t)} \quad (2b)$$

$$h_i^{(t+1)} = \sum_{i \to j} h_j^{(t)} \quad (2c)$$

$$a_i = norm(a_i), \quad h_i = norm(h_i) \quad (2d)$$

In the above equations, the values of an (Authority) and h (Hub) take 0 and 1 respectively as initial values (Eq. 2a). It is not necessary to give a value to Authority, 0 is given to identify nodes that do not have any connection from another node. After the initial values are given, the Authority value is first calculated for each node (Eq. 2b). After calculation of the entire network, Hub values are calculated for all nodes of the network (Eq. 2c). After Authority and Hub values are calculated, normalization is performed (Eq. 2d).

2.2.3. Betweenness Centrality

The Betweenness Centrality algorithm (Freeman, 1977) weights a node according to betweenness in the list of the shortest paths among all the nodes in the network. A node gets its weight with the ratio of existing in the shortest paths between all nodes. Betweenness centrality weight of a node can be calculated as in (Eq. 3):

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (3)$$

, *where* σ_{st} is the total number of shortest paths from node *s* to node *t*, and $\sigma_{st}(v)$ is the number of shortest paths from node *s* to node *t* that pass through from node *v*. To calculate the centroid weight of a node, it is necessary to calculate the shortest paths of each node to the other nodes. Due to these operations, according to other algorithms, it needs a considerable processing time for networks with a large number of nodes and relations. With new studies, its derivatives are proposed where intensive calculations can run faster (Brandes, 2001; Brandes and Pich, 2007).

2.2.4. Mentions-Sense Method

The Mentions-Sense method is the centrality measurement proposed by (Veronis and Ide, 1990) is used to calculate the centrality of the lemma nodes in the semantic network created from Turkish dictionary data. It emphasi(zes the MENTIONS relations of the lemma nodes, if a lemma node has many MENTIONS relations it is semantically a common word that exists in the definitions of the lemmas. And, on the other hand, if a lemma has more than one sense, the number of MENTIONS relations that connected to it must be divided into the number of senses (Eq. 4).

$$Centrality(L) = \left\lfloor ln\left(\frac{|L_{MENTIONS}|}{|L_{SENSES}|} + 1\right)\right\rfloor \quad (4)$$

Centrality(L), is the centrality value of the lemma node, $L_{MENTIONS}$ is the number of definitions referring to that lemma, is divided by the total number of sense nodes of that lemma L_{SENSES}, 1 is added to assure that the value is 0 or positive number, and then taking the natural logarithm of the result to normalize the centrality values of the nodes.

3.Results and Discussion

A semantic network created from a Turkish dictionary is modified to evaluate all 4 algorithms. In the modified version, there are only lemma nodes and one relation type named “RELATED”, in total 80,173 nodes with 629,289 relations. The synonym pairs that are extracted from Turkish dictionary data are used for comparison. The Lemma nodes of synonym pairs can have direct connections between each other, or they can be linked to each other over multiple nodes and relationships. A total of 78,964 shortest paths were queried between 39,482 synonym pairs in both directions. 72,353 shortest paths were found on the semantic network while 6,611 shortest path queries were returned without any path. Synonyms are expected to have a path between them in both directions, but not all possible paths can be found due to a deficiency of dictionary data. These 72,353 shortest paths are used to calculate the semantic distance between each synonym pairs by adding weights produced by the centrality algorithms.

In the Equations 5a-d, distance formulas are given for the algorithms, PageRank, HITS, Betweenness Centrality and Mentions-Sense, respectively. Here, *Distance* (L_1, L_2) is the semantic distance between synonym lemma nodes L_1 and L_2 and *P* is the shortest path between the synonym pair while *L* is a node in the shortest path *P*. PageRank algorithm is used with damping factor value 0.85 as provided in the paper of Brin and Page (1998) for Equation 5a.

$$Distance(L_1, L_2) = \sum_{L \in P} PR(L) \quad (5a)$$

$$Distance(L_1, L_2) = \sum_{L \in P} L_a * L_h \quad (5b)$$

$$Distance(L_1, L_2) = \sum_{L \in P} g(L) \quad (5c)$$

$$Distance(L_1, L_2) = \sum_{L \in P} Connectivity(L) \quad (5d)$$

The aforementioned algorithms are used to calculate the semantic relation of the 78,964 synonym pairs. At first, the two node weight values and the shortest path between each pair are calculated with each algorithm. Following that, the weight values of each node in the path are added up as the total score.

When the performance and scalability of the algorithms are considered, recurrent characteristics of the PageRank and HITS algorithms can be seen as a serious problem. Node/relation addition/deletion operations result in recalculation of the semantic graph. It is already known that the Betweenness Centrality algorithm is computationally complex. The Mention-Sense algorithm, in its non-recurrent and scalable structure, is cost-efficient when the other three algorithms are considered. This may be because of the bipartite architecture of the semantic graph. In Table 1, the score values of some selected synonym pairs are given for each algorithm.

Table 1. Similarity scores of two homonyms and their senses for word "kurt" on the semantic network

Word 1	Word 2	Mention-Sense	PageRank	HITS	Betweenness Centrality
vurmak	sürmek	0,4157	0,0059	1,1667	1,5714
sürmek	vurmak	0,5661	0,0101	1,8333	2,3810
hükümdar	kral	0,5832	0,0012	0,7738	1,2857
kral	hükümdar	0,8510	0,0040	1,2738	2,0000
seyrekleştirmek	aralamak	0,2920	0,0002	0,5952	1,3333
aralamak	seyrekleştirmek	0,2920	0,0002	0,5952	1,3333
katılan	müdahil	0,5558	0,0030	0,3036	1,1429
müdahil	katılan	0,5558	0,0030	0,3036	1,1429
bildirme	deklarasyon	0,7782	0,0042	1,1667	2,1905
deklarasyon	bildirme	0,4726	0,0034	0,5893	1,0476

In Table 1, it can be seen that HITS and Mention-Sense algorithms perform in a balanced score range but PageRank is too skewed. For example, for the synonym pair “aralamak-seyrekleştirmek” the PageRank score is 0,0002 while for the pair “müdahil-katılan”, the value is 0,0030. Sometimes the pair values for a pair can be different. For example, for the pairs, “deklarasyon-bildirme” and “bildirme-deklarasyon”, the scores are 0,4726 and 0,7782, respectively. The difference is caused by the word “bildirme” word that is found in the sense definition of the word “deklarasyon”, while the word “deklarasyon” cannot directly found in the definition of “bildirme” and more hops are needed in the semantic graph. In Table 2, the standard deviations of the four algorithms are given for the calculation of the synonym pairs after normalization.

Table 2. Comparison of the four graph weighting algorithms

	Mention-Sense	**PageRank**	**HITS**	**Betweenness Centrality**
Mention-Sense	0	0,7991	0,4138	0,9243
PageRank	0,7991	0	1,1437	1,6497
HITS	0,4138	1,1437	0	0,5936
Betweenness Centrality	0,9243	1,6497	0,5936	0
Mean of Deviation	**0,5343**	**0,8981**	**0,5378**	**0,7919**

In Table 2, it can be seen that PageRank has the highest mean of deviation from other algorithms' results, while the Mention-Sense algorithm has the lowest mean of deviation and again HITS and Mention-Sense algorithms perform near weight values for each pair. On the other hand, when a new node is added to the test semantic graph, all weight values should be re-calculated for PageRank, Betweenness Centrality and HITS algorithms. Eventually, the Mention-Sense algorithm generates similar weights to other algorithms while using less computation without using the entire network.

In Figure 4, the score distribution of the synonym pairs for each algorithm is given. All scores are sorted ascending for easy view.

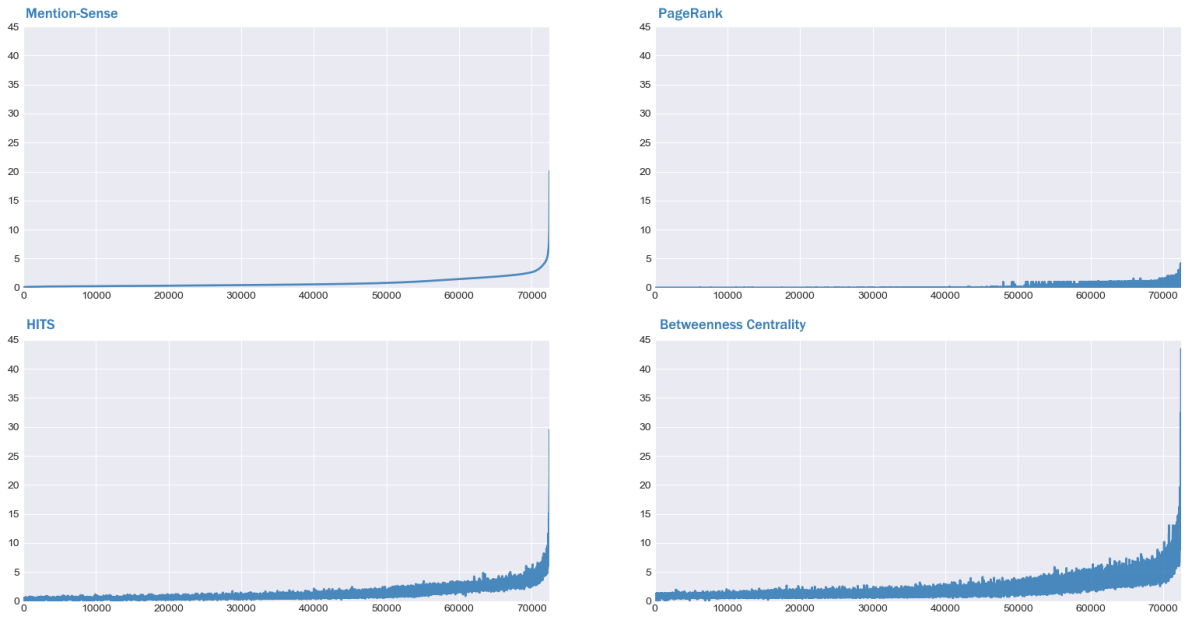

Figure 4. Synonym value distribution for the four algorithms

In Figure 4, the horizontal axis denotes the sorted index value of the synonym pairs while the vertical axis represents the score. It can be seen that; the trends are similar except for the ranges. Especially, Mention-Sense and HITS algorithm trends are very similar. In Figure 5, histograms of the scores, given by each algorithm can be seen.

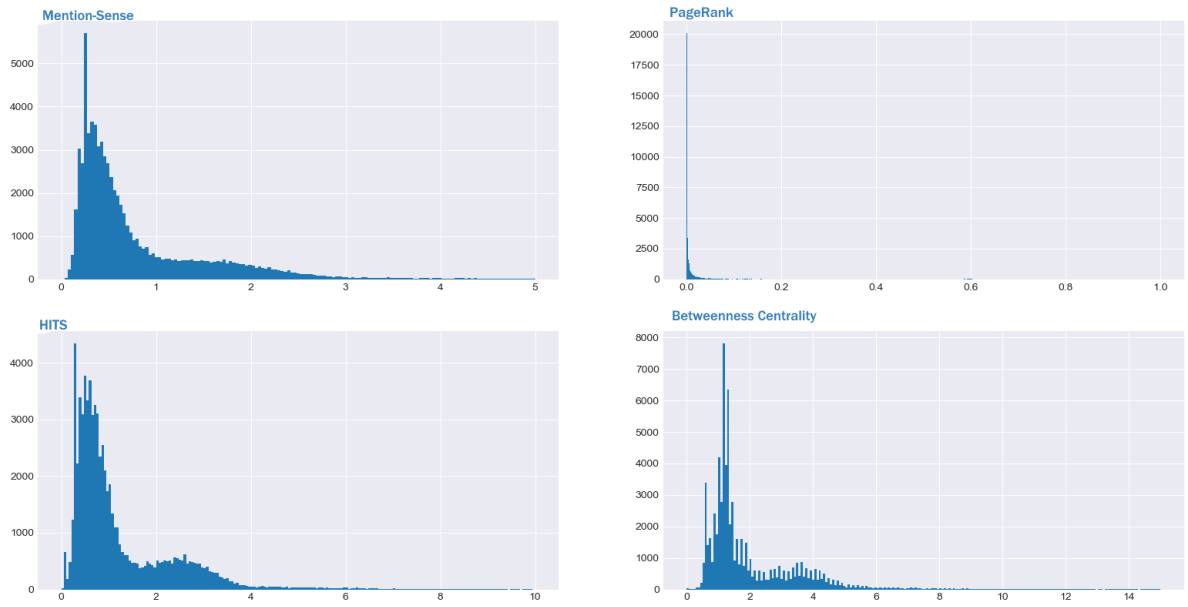

Figure 5. Synonym pair value histograms of the four algorithms

In Figure 5, the vertical and horizontal axes represent the frequency values and scores, respectively. The histograms of the semantic distance scores are in ranges between [0, 5] for Mention-Sense and HITS, [0,1] for PageRank and [0,10] for Betweenness Centrality methods, respectively. For the PageRank algorithm, the histogram of the weights appears as stuck in a range very close to 0, due to very low values. Three algorithms perform very near distribution, except PageRank. Again, as seen in Figure 4, Mention-Sense and HITS algorithm trends are very similar.

4. Conclusions and Suggestions

These three centrality algorithms are proven and robust algorithms in the literature of graph algorithms, but they have their setbacks when it comes to computational and applicability requirements on a bipartite graph.

Evaluations of four algorithms demonstrate that all of the algorithms generate weights close to each other except the PageRank algorithm. And our method, Mention-Sense, has the lowest mean of deviation to other algorithms' generated weights.

The simplicity of our method also provides faster weights generated locally with the adjacent nodes in the semantic network, while the algorithms have to use the entire graph to calculate the weights of a particular node. Our method requires less computation and process time, and also when a node is removed or a new node is added to the semantic network, re-calculations of weights are only applied to the adjacent nodes of removed or added nodes.

Acknowledgments

This study is a part of the research program with project number 215E256, which is financed by the Scientific and Technological Research Council of Turkey (TUBITAK).

Makale, araştırma yayın etiğine uygun olarak hazırlanmıştır. Yazarlar arasında çıkar çatışması yoktur.

References

Brin, S., Page, L., 1998. The anatomy of a large-scale hypertextual web search engine.

Brandes, U., 2001. A Faster Algorithm for Betweenness Centrality. Journal of Mathematical Sociology 25 (2): 163–77.

Brandes, U., Pich, C., 2007. Centrality Estimation in Large Networks. International Journal of Bifurcation and Chaos 17 (7): 2303–18.

Freeman, L.C., 1977. A set of measures of centrality based on betweenness. *Sociometry*, 35-41.

Gleich, D.F., 2015. PageRank Beyond the Web. Siam Review 57 (3): 321–63.

Kleinberg, J.M., Kumar, R., Raghavan, P., Rajagopalan, S., Tomkins, A.S., 1999. The web as a graph: measurements, models, and methods. In *International Computing and Combinatorics Conference* (pp. 1-17). Springer, Berlin, Heidelberg.

Manaskasemsak, B., Rungsawang, A., 2005. An Efficient Partition-Based Parallel PageRank Algorithm. In 11th International Conference on Parallel and Distributed Systems (ICPADS'05), 1:257–63.

Miller, G.A., 1995. WordNet: a lexical database for English. *Communications of the ACM*, *38*(11), 39-41.

Li, W., Liu, C. C., Zhang, T., Li, H., Waterman, M.S., Zhou, X.J., 2011. Integrative analysis of many weighted co-expression networks using tensor computation. *PLoS Comput Biol*, *7*(6), e1001106.

Newman, M., 2018. Networks. 2nd Ed. Oxford University Press.

Savić, M., Ivanović, M., Jain, L.C., 2019. Fundamentals of Complex Network Analysis. In: Complex Networks in Software, Knowledge, and Social Systems. Intelligent Systems Reference Library, vol 148. Springer, Cham. https://doi.org/10.1007/978-3-319-91196-0_2

Tibermacine, O., Tibermacine, C., Kerdoudi, M.L., 2019. Reputation Evaluation with Malicious Feedback Prevention Using a HITS-Based Model. In 2019 IEEE International Conference on Web Services (ICWS), 180–87.

Turan, E., Orhan, U., 2018. Building a Turkish Semantic Network and Connecting Synonym Senses Bidirectionally. In *2018 Innovations in Intelligent Systems and Applications (INISTA)* (pp. 1-6). IEEE.

Veronis, J., Ide, N., 1990. Word sense disambiguation with very large neural networks extracted from machine readable dictionaries. In *COLNG 1990 Volume 2: Papers presented to the 13th International Conference on Computational Linguistics*.