



Node Ordering for Rescalable Network Summarization (or, the Apparent Magic of Word Frequency and Age of Acquisition in the Lexicon)

Violet Brown¹, Xi Chen², Maryam Hedayati^{1,2}, Camden Sikes², Julia Strand¹, Tegan Wilson², and David Liben-Nowell²(✉)

¹ Department of Psychology, Carleton College, Northfield, MN 55057, USA

² Department of Computer Science, Carleton College, Northfield, MN 55057, USA
dln@carleton.edu

Abstract. How can we “scale down” an n -node network G to a smaller network G' , with $k \ll n$ nodes, so that G' (approximately) maintains the important structural properties of G ? There is a voluminous literature on many versions of this problem if k is given in advance, but one’s tolerance for approximation (and the resulting value of k) will vary. Here, then, we formulate a “rescalable” version of this approximation task for complex networks. Specifically, we propose a *node ordering* version of graph summarization: permute the nodes of G so that the subgraph induced by the first k nodes is a good size- k approximation of G , averaged over the full range of possible sizes k . We consider as a case study the phonological network of English words, and discover two natural word orders (word frequency and age of acquisition) that do a surprisingly good job of rescalably summarizing the lexicon.

Keywords: Network summarization · Node ordering
Phonological networks

1 Introduction

At SIGGRAPH 2007, Shai Avidan and Ariel Shamir presented a remarkable technique for “content-aware image resizing” [5]: shrink the size of an image while preserving, to the greatest extent possible, its important visual qualities. This problem can be solved crudely by simple cropping or rescaling, but Avidan and Shamir’s approach is more subtle: they identify an ordering of the “seams” (contiguous edge-to-edge paths through the image) from least important to most

This work grew out of portions of a research project that was carried out by the authors of this work in collaboration with Aman Panda and Duo Tao. We gratefully acknowledge their contributions. This work was supported in part by Carleton College. Comments are welcome.

© Springer Nature Switzerland AG 2019

L. M. Aiello et al. (Eds.): COMPLEX NETWORKS 2018, SCI 812, pp. 66–80, 2019.

https://doi.org/10.1007/978-3-030-05411-3_6

important. Their algorithm allows a user to shrink an image from n pixels on a side to any size $k \leq n$, with k chosen by the user in real time, by eliminating the $n - k$ least important seams.

What would it mean to perform an analogous “resizing” for a complex network? Is there a meaningful way to shrink an n -node network to any size $k \leq n$, with k chosen on the fly by the user—say, ordering the nodes by “representativeness”—while preserving important graph-theoretic qualities, so that the graphs in the resulting nested sequence are “as much like” the original as possible?

Approximating complex networks. When members of the complex-networks community describe a network as “complex,” we seem to have in mind a fuzzy constellation of properties, expecting the network to exhibit many of these desiderata: e.g., “small-world” properties [49], a heavy-tailed degree distribution [12], community structure [17], and degree assortativity [38]. We probably also expect the network to be “large.” (Despite the now-ostentatious attention to Zachary’s Karate Club [51], few researchers would argue for it as a paradigmatic complex network; how much complexity can 34 nodes admit?) For a variety of reasons, though, the large size of a network can be problematic. This issue is immediate in the sense of computational complexity—one cannot afford $\Omega(n^2)$ time on a billion-node social network—and it is even more of an issue if one seeks some kind of real-world intervention.

With these sorts of motivations in mind, many researchers have performed significant work on the task of taking a large complex network and performing a type of lossy compression on it; that is, identifying some smaller graph (either by deleting or aggregating nodes) that is a useful approximation to the original. But this problem is difficult for a number of reasons: algorithms for the network approximation problem itself often have running times that grow unfavorably in the size of the full network; the resulting smaller network may vary widely depending on the size of the desired subnetwork; and it is unclear as to the right way to assess the quality of the smaller subnetwork. (See [1,33,34] and Sect. 3.)

The present work: node ordering as (rescalable) network summarization. The goal of these graph summarization algorithms is to preserve “interesting” properties of the graph, while reducing the size of the graph as much as possible. But a major challenge here—highlighted clearly by Liu et al. [34]—is that what counts as “interesting” will differ from one researcher to another (and, for that matter, so will what counts as “preserved”). And size-reduction algorithms may well require us to precommit to the size of the desired smaller network *and* to the network properties of interest, both of which may be undesirable. (Though see [35].)

Here, we propose a task that embraces these differences in the desired level of approximation: given a complex network G , we seek to identify an *order* v_1, v_2, \dots, v_n of the nodes of G such that the “prefix graph” for a given size k —that is, the subgraph induced by the node set $\{v_1, v_2, \dots, v_k\}$ —is as close

an approximation to G as possible, for any desired size k . We quantify success for this *Node Ordering Problem* in both the sense of global statistics about the graph, and local statistics about the importance of individual nodes in the full graph and the subgraphs.

We will focus on a particular complex network as a case study for our discussion: the *phonological network*, in which nodes correspond to words in a natural language (here, English), and edges connect pairs of nodes whose pronunciations differ by a single edit [4, 43, 47, e.g.]. We will describe some natural node orderings in this network, including two derived from external data sources—word frequency and age of acquisition—that do a remarkably good job of “unkinking” the phonological network, producing a nested sequence of graphs that reproduce to a surprising extent the statistical properties of the lexicon as a whole.

2 The Node Ordering Problem: Approximating Degree

Our framework of successively approximating a graph via one-by-one additions of nodes is quite broad; we could apply it with a variety of graph-theoretic quantities, and also with a variety of ways to quantify the difference between two graphs with respect to any particular quantity. But, to start, we will formalize one specific version of the Node Ordering Problem, using what is perhaps the simplest nontrivial way to compare graphs: the *degree* of the graphs’ nodes. Let $\delta(u, G)$ be a function reporting the degree of the node u in any graph G .

Prefix graphs. First, we fix a bit of terminology. We are given an undirected graph $G = \langle V, E \rangle$, called the *full network*. Denote by $n = |V|$ the number of nodes in G . We will refer to a permutation of the vertices $\pi = \pi_1, \pi_2, \dots, \pi_n$ as a *node ordering*.

Any particular permutation π defines a sequence of n *prefix networks*, one of each size between 1 and n ; specifically, the k -node prefix network of G under π is the subgraph of G induced by the nodes $\{\pi_1, \pi_2, \dots, \pi_k\}$. (The *subgraph of $G = \langle V, E \rangle$ induced by a set $A \subseteq V$* is the graph G_A with nodes A and containing all edges in E that join two nodes in A ; that is, $G_A = \langle A, \{(u, v) \in E : u \in A \text{ and } v \in A\} \rangle$.)

See Fig. 1 for a small example. (Note that, as always, the last prefix network *is* the full network—i.e., in Fig. 1, we have $G = G_{\{1,2,3,4\}}$.)

Measuring structural quality of a subnetwork. Any node ordering defines a nested sequence of prefix graphs, starting with a single isolated node and ending with G itself. We must now describe the objective function—i.e., how do we assess the quality of a particular permutation? Our evaluation is guided by three principles. First, we seek *low discrepancy* between the sequence of prefix graphs and the full network (averaged over all n different prefix sizes). Second, we measure discrepancy using *relative error*: if the prefix graph exhibits a value x and the full network a value x^* , then we compute the error as $\frac{|x-x^*|}{x^*}$.

Third, we want to capture both *global error* (does the prefix graph have similar average statistics to the full network?) and *local error* (do those nodes that appear in the prefix graph have similar statistics there as they do in the full network?). In keeping with these principles, we define two notions of error:

Definition 1 (Global Error). For a prefix graph G_A , the global (relative) degree error is the relative error of the mean degree of G_A compared to that of $G = \langle V, E \rangle$:

$$\text{global error of } G_A = \frac{\left| \frac{1}{|A|} \cdot \left[\sum_{v \in A} \delta(v, G_A) \right] - \frac{1}{|V|} \cdot \left[\sum_{v \in V} \delta(v, G) \right] \right|}{\frac{1}{|V|} \cdot \left[\sum_{v \in V} \delta(v, G) \right]}.$$

Definition 2 (Local Error). For a prefix graph G_A , the local (relative) degree error is the mean relative error of each node in G_A compared to $G = \langle V, E \rangle$, averaged across all nodes present in G_A :

$$\text{local error of } G_A = \frac{1}{|A|} \cdot \sum_{v \in A} \frac{|\delta(v, G_A) - \delta(v, G)|}{\delta(v, G)}. \tag{1}$$

The errors for the prefix graphs for our small sample graph are also given in Fig. 1.

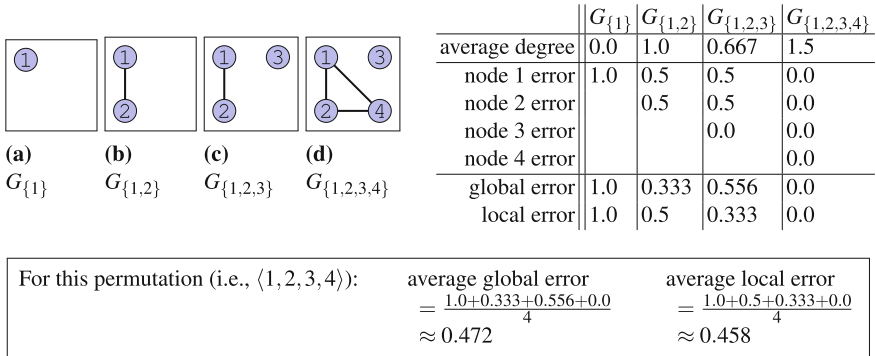
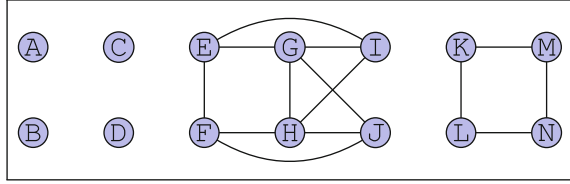


Fig. 1. Consider the full network G with nodes $\{1, 2, 3, 4\}$ and edges $\{(1, 2), (2, 4), (1, 4)\}$, under the permutation $\langle 1, 2, 3, 4 \rangle$. First, we show the four prefix graphs for this network under this permutation. Then, we show the stage-by-stage local and global errors. (For example, node 2’s local error in $G_{\{1,2\}}$ is 0.5 because it has degree 1 in $G_{\{1,2\}}$ and degree 2 in G , so its local error is $|2 - 1|/2 = 0.5$.) Note that average global and local error are permutation dependent, so these values would be different for a different ordering of the nodes.

Note that global and local error measure different things: low global error corresponds to the *density* of G_A matching that of G , while low local error corresponds to specific node degrees in G_A matching their degrees in G . It is possible for a graph G_A to have low global error while simultaneously having high local error (i.e., this graph maintains the average degree of G , but the average arises from different local connections among nodes in the two graphs), or vice versa. See Fig. 2.



	ABCD	EFGH	KLMN	EJKN
global error	1.0	0.0	0.0	1.0
local error	0.0	0.417	0.0	1.0

Fig. 2. A graph with 14 nodes and 14 edges, and four different sets of four nodes whose induced graphs exhibit all four combinations of high/low local error and high/low global error.

Here, then, is the formal statement of our problem:

Definition 3 (Node Ordering Problem [Degree Version]). *Given an undirected graph $G = \langle V, E \rangle$, output the permutation of V that minimizes the average total error (global + local), where the average is taken across all $|V|$ prefix graphs.*

Although we have focused on degree as the node-level measure of interest, all of our definitions apply for an arbitrary node-level function. Many other measures are at least as interesting to consider as degree—but even this “simple” measure will reveal some surprisingly complex and subtle network features.

3 Related Work

Here we will (nonexhaustively) highlight some of the work in the many areas of related research. First, though, we note that our Node Ordering Problem is fundamentally different from ranking the nodes by some kind of centrality measure, in which the most important nodes appear earliest; rather, we are trying to produce an order of the nodes that is “always” roughly as important as the list as a whole; how good a node is to include next depends on which nodes have already been selected.

Summarizing and sampling in networks. The most closely related body of research—and also the most voluminous—studies algorithms to shrink large complex networks. Closest is work on *simplifying* graphs through the removal of nodes (or, less similarly, edges) [19, 40, 46]. There is also a great deal of work on *sampling* graphs, in which one tries to choose a good set of representative nodes from a large network on the fly, generally without knowing the full graph [20, 30, 35, 36, 39]. The excellent surveys of Liu et al. [34], Lin et al. [33], and

Ahmed et al. [1] describe much more of this line of work—beyond the highly incomplete list cited here—including research on other kinds of graph compression (e.g., the aggregation of many nodes into supernodes) that are further afield from our task.

Modeling the evolution of graphs. In Sect. 4, we examine how a particular graph G evolves—namely, how the phonological network changes as we add words, one by one, in the order that an average person learns them. When nodes are ordered by arrival time, the prefix graphs form a flipbook of G ’s temporal evolution. Questions about how particular complex networks evolve over time are well studied, ranging from the local (which new links will form, and when?) [8, 29, 32] to the global (how will density and diameter change over time?) [31]. That work generally considers *both nodes and edges arriving over time*; here, we “know the future” of the network—the edge (u, v) forms at precisely the moment that the second of the two nodes arrives in the graph—so the kind of graph evolution that we see is generally quite different from the changes studied in this literature.

Graph drawing, minimum linear arrangement, and comparing permutations with costs varying depending on position. Multiple computational communities (from graph drawing to VLSI design) have considered the task of ordering the nodes of a graph so that edges connect nodes at nearby positions in the ordering. In the graph-drawing context, the resulting images are called *arc diagrams* [37, 48]; ordering nodes to minimize the total length of edges in an arc diagram is called “minimum linear arrangement (MinLA)” [14, 16], which is NP-hard. (This problem is also similar to that of approximating a general metric as a line [15].)

MinLA is a close match for our notion of local error. In MinLA, one seeks to minimize $\sum_{(u,v) \in E} |\pi_u - \pi_v|$; here, the number of prefix graphs in which exactly one of $\{u, v\}$ appears is precisely $|\pi_u - \pi_v|$. But there are important differences: we consider relative, not absolute, error, and we average error across all prefix graphs rather than summing error over edges; a single node’s local error counts less when there are more nodes in the ordering (because that cost is divided by a larger population size). In most ordering problems, as in MinLA, the cost measure does not depend on the location of any errors, though a few researchers have recently studied scenarios that, like ours, penalize errors differently depending on where the error sits [21, 25].

4 Case Study: Word Recognition and the Phonological Network

In this section, as a case study, we consider a particular medium-sized complex network. Our network comes from the psycholinguistics literature on *spoken-word recognition* [4, 47]: nodes represent the words in the language, and we join two words w and w' by an edge if their pronunciations differ only by a single phonemic insertion, deletion, or substitution. For example, neighbors of *cowl* /kaʊl/

include *scowl* /skaʊl/ (an insertion), *owl* /aʊl/ (a deletion), and *fowl* /faʊl/ (a substitution). This network has many of the properties that we discussed previously: a giant component, small average path lengths, high clustering coefficients, degree assortativity, etc. [4, 43, 47]. We obtained our list of words and pronunciations from the English Lexicon Project [6]. We discarded words for which we had no *word frequency* or *age of acquisition* data (see below), and removed homophones, keeping only the highest frequency word with each pronunciation. The resulting graph G_{lex} contains $n = 30,515$ words, with an average degree ≈ 3.5 .

Ordering nodes randomly. As a baseline, we begin with a *random* ordering of the nodes of G_{lex} . Figure 3 shows both global and local error rates for 16 random orderings of the words in G_{lex} , as the fraction of nodes included in the graph ranges from 0 to 100%. To calculate a single measure of the quality of each order π , we compute the average error rates across all n prefix graph sizes of π , resulting in a pair of numbers per ordering. (We calculate this average approximately, averaging the error for prefix graphs of size 100, 200, \dots , n , and round to the hundredths place.)

When the nodes are ordered randomly, we see a linear trend for both global and local error, as we could expect. Let $G[\alpha]$ denote the prefix graph resulting from including a random α -fraction of the nodes of G_{lex} . The global error of $G[\alpha]$ is the fraction by which $G[\alpha]$'s average degree is lower than G_{lex} 's average degree; in expectation, this fraction drops linearly with α . (A particular edge from G_{lex} is included in $G[\alpha]$ with probability $\approx \alpha^2$ —both endpoints must appear in $G[\alpha]$ —so $G[\alpha]$ will contain $n \cdot \alpha$ nodes and, on average, $|E| \cdot \alpha^2$ edges, and thus an average degree of $(2|E|\alpha^2)/(n\alpha) = (2|E|/n) \cdot \alpha$. The average degree of G_{lex} is $2|E|/n$.) Average local error is similarly linear, though starting at about 0.7 instead of

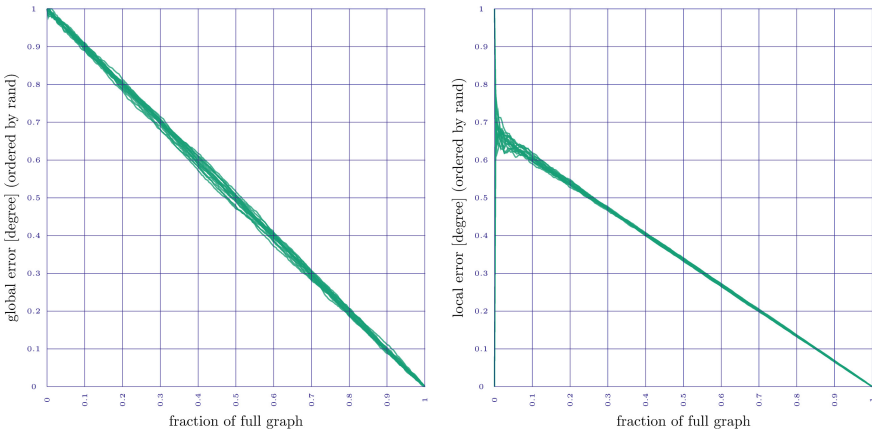


Fig. 3. Global error (left) and local error (right) rates for 16 random orderings of the words in the lexicon. In these orders, the average global error was always in $\{0.49, 0.50, 0.51\}$ (median = 0.50); local error was always 0.33 or 0.34 (median = 0.34).

1.0: about 30% of the nodes in G_{lex} are isolated (i.e., have no neighbors); these nodes have zero local error at the very moment that they are added into the graph.

Ordering nodes by network properties. To minimize local error for a node u as quickly as possible, we want as many edges incident to u to appear in the graph as soon after u in the order as possible. Adding nodes greedily by degree, highest degree first, tends to achieve this goal, because G_{lex} is assortative with respect to degree. Thus ordering nodes by degree, breaking ties randomly, seems promising. While this greedy-by-degree order is good for local error, though, it does poorly with global error: the average degree rapidly shoots far above G_{lex} 's 3.5, and stays well above that target for almost all prefix sizes. Greedy-by-degree ordering yields local error 0.10 and global error 1.19 (median across 16 different random tiebreakers), substantially worse than the random ordering. (Note that relative errors exceed one when the prefix graph's average degree is more than twice that of G_{lex} .)

We find a similar effect when we order the nodes by *closeness* or *betweenness centrality*: the most central nodes' degrees are too high, and the global average goes, and stays, too high (closeness global error 1.18, local 0.10; betweenness global error 0.85, local 0.17). We could also greedily add nodes by degree, *lowest* first; this strategy has an analogous problem, but with a persistently too-low global average.

Ordering nodes by external properties. Indeed, it is hard to formulate a network-theoretic property that would intuitively yield good performance. Somehow we need a sequence of nodes in which we tend to add “regions” of the graph at a similar time (so that newly added nodes' local error drops quickly), while also ensuring that those regions have nodes that are typical of the whole graph (so that the global error stays low). Of course, one could explicitly select for these desired properties—e.g., repeatedly greedily removing the node whose removal increases total error by the least—but here we consider another option: ordering the nodes by psycholinguistic properties that are, at least nominally, independent of graph position:

- *Frequency.* We obtained word frequency counts for all the words in our lexicon from the SUBTLEX_{US} corpus of 51 million words of American subtitles [9], stored as frequency per million words.
- *Age of acquisition.* We used ratings of the age at which a given word was learned, its *age of acquisition* (AoA), from Kuperman et al. [26]. These data were obtained by adults retrospectively self-reporting the age at which they learned a given word; the data are expanded so that w 's AoA is recorded based on the “lemma” of w —e.g., *endorsed* is recorded as being acquired at the same age as *endorse*. Despite the inherent limitations of such self-reporting, Kuperman et al. [26] argue that these estimates accurately reflect the order in which words were learned, and the data have proven predictive in other psycholinguistic settings [10, 13].

Although higher frequency words tend to be acquired earlier, these two quantities capture different phenomena, particularly for the lemma-expanded version of AoA. Many pairs of words (29.5%) are inverted in the AoA vs. frequency orders (e.g., *water* is early in both lists, *watered* is early AoA [because its lemma is *water*] but low frequency, and *business* is high frequency but acquired fairly late).

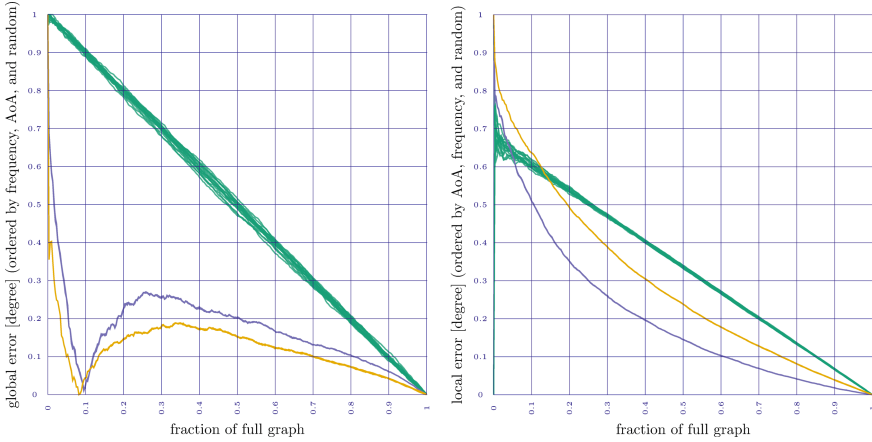


Fig. 4. Global (left) and local (right) error rates for degree for three word orders: random (green), frequency [high to low] (gold), and AoA [low to high] (blue). Frequency has the lower global error and AoA the lower local error; both are better than random in both measures.

The analogue to Fig. 3 for these two orderings is shown in Fig. 4. (We resolve any ties in the ordering by randomizing, executing 16 distinct runs for each measure.) We see notable improvement in the global error over the random ordering: the average global error for frequency is 0.12, and for AoA is 0.17 (vs. 0.50 for random). For local error, the difference is less pronounced, but error is still smaller than that for the random ordering: frequency’s average local error is 0.29 and AoA’s is 0.21 (vs. 0.34 for random). Note that the points at which AoA and frequency’s global error first hits zero, at $\approx 10\%$ of the full graph, are the points at which the prefix graphs’ average degree first exceeds G_{lex} ’s average: to the left of that point, the prefix graphs are too sparse; to the right, the prefix graphs are too dense.

We also tried ordering the nodes by two other properties that are nominally unrelated to network position: in increasing order of *orthographic length* (how many letters are in the spelling of the word?) and *phonological length* (how many phonemes are in the pronunciation of the word?). These orderings suffer from the same problem as ordering by centrality: the fraction of possible k -phoneme strings that are actually words decreases with k , so short words have many more

neighbors than average; the global error for word-length orderings is quite high (> 0.8) as a result.

Going beyond degree: clustering coefficient. Although we introduced it strictly in the context of degree, we can consider versions of the Node Ordering Problem for any node-level property. Here, we consider *clustering coefficient*: the fraction of pairs of a node’s neighbors that are directly joined by an edge. (Clustering coefficient of G_{lex} has been studied in several psycholinguistic contexts [2, 11, 50].) See Fig. 5: frequency and AoA both vastly outperform random ordering for global error, and are roughly comparable in local error. (Figure 5 shows that they achieve this local error in different ways, though: the random ordering benefits from the fact that about 65% of words have degree ≤ 1 , and ergo clustering coefficient = 1. Frequency and AoA tend to do poorly on their early prefix graphs, before overtaking the random ordering about a third of the way through the graph.)

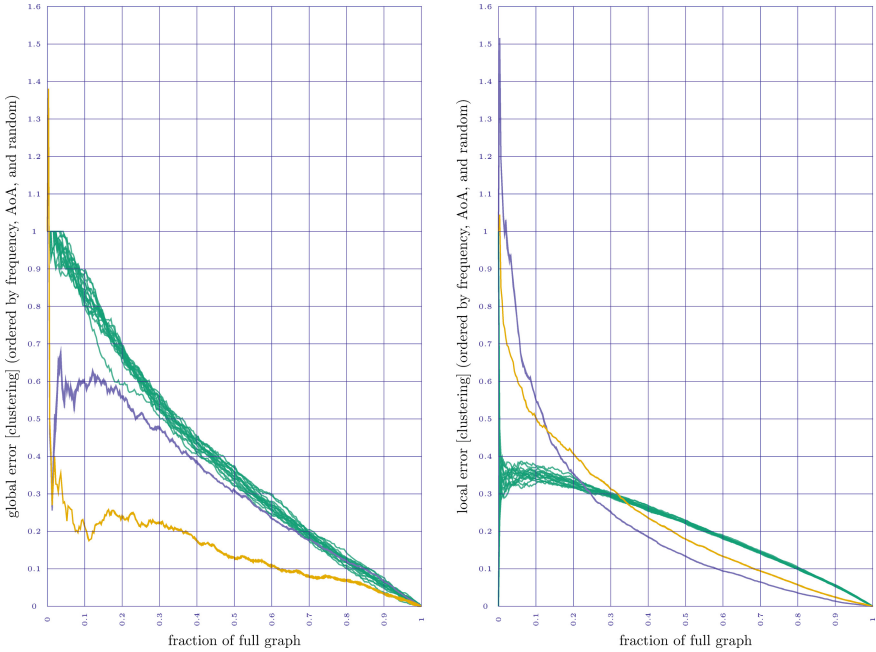


Fig. 5. Global (left) and local (right) error rates for clustering coefficient for words ordered randomly (green), by decreasing frequency (gold), and increasing AoA (blue). All other orders mentioned previously (degree, betweenness, closeness, and word length) have worse average total error (global+local) than random ($0.40+0.21$); both frequency ($0.15+0.23$) and AoA ($0.32+0.22$) are again better than random.

5 Discussion and Future Directions

The task that we introduced in this paper is a broad one: order the nodes of a given complex network in a permutation π such that the prefix graph induced by $\{\pi_1, \pi_2, \dots, \pi_k\}$ is a good (global and local) approximation to the full network, averaging over the possible values of k . There are, of course, a slew of ways to measure the similarity of the prefix graph and the full network, many more than the degree and clustering coefficient measures that we examined here. Understanding the extent to which a good node order for these two measures is also a good node order for other key graph-theoretic properties is an obvious next step. Many of the summarization and sampling algorithms for complex networks (see Sect. 3) could potentially be adapted to this setting, too.

For any particular fixed node-level property, there is a natural greedy algorithm that applies: starting from the full graph, repeatedly put at the end of the node order that vertex whose removal increases total error by the least. Hill-climbing algorithms could also be adapted fairly straightforwardly to this setting. These approaches differ from the measure-agnostic view of the ordering task that we have taken so far (we seek a node order that “in all important ways” reproduces the full graph), but one may approach the problem from a measure-specific perspective (see [35]). This style of algorithm may be computationally prohibitive, though; incremental algorithms for the measure in question (e.g., [28, 41]) are necessary, but not sufficient, to make the computation feasible.

Node ordering in the phonological network. The most salient fact from our examination of the phonological network is that ordering words by frequency or age of acquisition results in remarkably low error, both global and local, and that these orderings outperform the random baseline in both degree and clustering coefficient. Shorter words tend to appear early in these lists, and shorter words tend to have higher degree—but both frequency and AoA outperform degree-based and word-length-based orders for the nodes.

Why might AoA and frequency do such a good job in ordering the nodes of the phonological network for degree and clustering coefficient? In part, it seems, their success stems from a sense in which these word properties interpolate between two competing goods: soon after a node u appears in the order, we want to add many of u ’s neighbors (so that u ’s local error drops quickly), *but* we must avoid too much BFS-style exhaustive exploration of a dense “community” involving u (which would cause the global average, and thus the global error, to spike).

These two successful orderings generally do some BFS-style exploration around words as they are added: e.g., the frequency of *write* and an inflected form like *writes* are quite similar—and their lemma-expanded AoAs are exactly identical—so *writes* comes along soon after *write*. But AoA and frequency avoid immediately flooding the immediate neighborhood: most phonological neighbors of *write* are semantically (and thus morphologically) unrelated to writing, and therefore would not generally have a particularly similar frequency or AoA to *write*. (Note, then, that frequency is on the slightly more global side of the

global/local tradeoff, and AoA is on the slightly more local side. This observation is consistent with their local and global error rates.) Ordering nodes by degree does much worse than ordering by frequency, e.g., despite the positive correlation between degree and frequency [27], the degree order is too local in its exploration and thus suffers in global error.

Node ordering in other complex networks. There is a simple and more basic observation implied by the good results of AoA and frequency in ordering the nodes of the phonological network: that there *exists* some ordering of its nodes that “unkinks” its nodes in a way that leads to a sequence of good approximations to the network as a whole. That observation may say something important about AoA and frequency—or it may say something important about G_{lex} . Indeed, “unkinkability” may point to some extraordinarily odd features of the network. (Some recent research has begun to ask key questions about whether graph-theoretic properties of G_{lex} reflect interesting facts about English, or whether they are simply an artifact of the way that the network is constructed [18, 42, 44, 45].)

Perhaps the most compelling direction for further research on the node ordering problem is this: is there any meaningful analogue to Age of Acquisition in other kinds of complex networks? What happens if one tries to order the nodes of, say, a social network instead?

Although the superficial processes are quite different, after some reflection on the two just-discussed ways that node orders can perform poorly (being too local or not being local enough), an analogy between AoA and “social influence” begins to emerge. For example, in models of the spread of some behavior like the adoption of some new technology, that behavior can fail to spread widely by being too local (a small community adopts but it never spreads beyond that corner of the graph) or by not being local enough (adopters are too far apart, leading to isolated early adopters that have no common neighbors to jointly influence into adopting). It is an interesting open question as to whether ordering nodes by their order of adoption in, e.g., an Independent Cascade-style spread of behavior [22] (or perhaps the “backbone” of a network’s systemic communication lines [24]) might yield good performance.

Indeed, there are several real-world phenomena that seem to exhibit complex, high-dimensional behavior—and yet there is a way to unwind them into linear orders that approximate them remarkably well. This is true of postal codes in the United States, in which physical distance between locations is well approximated by the numerical difference in ZIP codes [3, 23]. It is also true of the web graph, in which links can be represented very efficiently if the underlying graph is stored with its nodes sorted lexicographically by URL [7]. Is there a way to linearly order the nodes of a social network, or indeed any other complex network, in a similar way?

References

1. Ahmed, N., Neville, J., Kompella, R.: Network sampling: from static to streaming graphs. *ACM Trans. Knowl. Discov. Data (TKDD)* **8**(2), 7 (2014)
2. Altieri, N., Gruenfelder, T., Pisoni, D.: Clustering coefficients of lexical neighborhoods: Does neighborhood structure matter in spoken word recognition. *Mental Lex.* **5**(1), 1–21 (2010)
3. Arbesman, S.: The fractal dimension of ZIP codes. *WIRED* (2012)
4. Arbesman, S., Strogatz, S., Vitevitch, M.: The structure of phonological networks across multiple languages. *Int. J. Bifurc. Chaos* **20**(03), 679–685 (2010)
5. Avidan, S., Shamir, A.: Seam carving for content-aware image resizing. *ACM Trans. Graph.* **26**(3), 10 (2007)
6. Balota, D., et al.: The English Lexicon project. *Behav. Res. Methods* **39**(3), 445–459 (2007)
7. Boldi, P., Vigna, S.: The webgraph framework I: compression techniques. In: *WWW 2004*
8. Brot, H., Muchnik, L., Goldenberg, J., Louzoun, Y.: Evolution through bursts: network structure develops through localized bursts in time and space. *Netw. Sci.* **4**(3), 293–313 (2016)
9. Brysbaert, M., New, B.: Moving beyond Kucera and Francis: a critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for American English. *Behav. Res. Methods* **41**(4), 977–990 (2009)
10. Brysbaert, M., Van Wijnendaele, I., De Deyne, S.: Age-of-acquisition effects in semantic processing tasks. *Acta Psychol.* **104**(2), 215–226 (2000)
11. Chan, K., Vitevitch, M.: The influence of the phonological neighborhood clustering coefficient on spoken word recognition. *J. Exp. Psychol. Hum. Percept. Perform* **35**(6), 1934–1949 (2009)
12. Clauset, A., Shalizi, C., Newman, M.: Power-law distributions in empirical data. *SIAM Rev.* **51**(4), 661–703 (2009)
13. Cortese, M., Khanna, M.: Age of acquisition predicts naming and lexical-decision performance above and beyond 22 other predictor variables: an analysis of 2,342 words. *Q. J. Exp. Psychol.* **60**(8), 1072–1082 (2007)
14. Devanur, N., Khot, S., Saket, R., Vishnoi, N.: Integrality gaps for sparsest cut and minimum linear arrangement problems. In: *STOC 2006*
15. Dhamdhere, K.: Approximating additive distortion of embeddings into line metrics. In: *APPROX/RANDOM 2004*
16. Feige, U., Lee, J.: An improved approximation ratio for the minimum linear arrangement problem. *Inf. Process. Lett.* **101**(1), 26–29 (2007)
17. Girvan, M., Newman, M.: Community structure in social and biological networks. *Proc. Natl. Acad. Sci.* **99**(12), 7821–7826 (2002)
18. Gruenfelder, T., Pisoni, D.: The lexical restructuring hypothesis and graph theoretic analyses of networks based on random lexicons. *J. Speech Lang. Hear. Res.* **52**(3), 596–609 (2009)
19. Hennessey, D., Brooks, D., Fridman, A., Breen, D.: A simplification algorithm for visualizing the structure of complex graphs. In: *INFOVIS 2008*
20. Hübler, C., Kriegel, H.P., Borgwardt, K., Ghahramani, Z.: Metropolis algorithms for representative subgraph sampling. In: *ICDM 2008*
21. Järvelin, K., Kekäläinen, J.: IR evaluation methods for retrieving highly relevant documents. In: *SIGIR 2000*

22. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: KDD 2003
23. Kosara, R.: US ZIPScribble map. <https://eagereyes.org/zipscribble-maps/united-states> (2006)
24. Kossinets, G., Kleinberg, J., Watts, D.: The structure of information pathways in a social communication network. In: KDD 2008
25. Kumar, R., Vassilvitskii, S.: Generalized distances between rankings. In: WWW 2010
26. Kuperman, V., Stadthagen-Gonzalez, H., Brysbaert, M.: Age-of-acquisition ratings for 30,000 English words. *Behav. Res. Methods* **44**(4), 978–990 (2012)
27. Landauer, T., Streeter, L.: Structural differences between common and rare words: failure of equivalence assumptions for theories of word recognition. *J. Mem. Lang.* **12**(2), 119 (1973)
28. Lee, M.J., Lee, J., Park, J.Y., Choi, R.H., Chung, C.W.: Qube: a quick algorithm for updating betweenness centrality. In: WWW 2012
29. Leskovec, J., Backstrom, L., Kumar, R., Tomkins, A.: Microscopic evolution of social networks. In: KDD 2008
30. Leskovec, J., Faloutsos, C.: Sampling from large graphs. In: KDD 2006
31. Leskovec, J., Kleinberg, J., Faloutsos, C.: Graph evolution: densification and shrinking diameters. *ACM Trans. Knowl. Discov. Data* **1**(1), 2 (2007)
32. Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.* **58**(7), 1019–1031 (2007)
33. Lin, S.D., Yeh, M.Y., Li, C.T.: Sampling and summarization for social networks (Tutorial). In: Pacific Asia Knowledge Discovery and Data Mining (2013)
34. Liu, Y., Safavi, T., Dighe, A., Koutra, D.: Graph summarization methods and applications: a survey. *ACM Comput. Surv.* **51**(3), 62 (2018)
35. Maiya, A., Berger-Wolf, T.: Benefits of bias: towards better characterization of network sampling. In: KDD 2011
36. Maiya, A., Berger-Wolf, T.: Sampling community structure. In: WWW 2010
37. Nagel, T., Duval, E.: A visual survey of arc diagrams. In: IEEE Visualization (2013)
38. Newman, M.: Assortative mixing in networks. *Phys. Rev. Lett.* **89**(20), 208,701 (2002)
39. Rafiei, D., Curial, S.: Effectively visualizing large networks through sampling. In: VIS 2005
40. Ruan, N., Jin, R., Huang, Y.: Distance preserving graph simplification. In: ICDM 2011
41. Sariyuce, A., Kaya, K., Saule, E., Catalyurek, U.: Incremental algorithms for closeness centrality. In: IEEE International Conference on Big Data (2013)
42. Shoemark, P., Goldwater, S., Kirby, J., Sarkar, R.: Towards robust cross-linguistic comparisons of phonological networks. In: Computational Research in Phonetics, Phonology, and Morphology (2016)
43. Siew, C.: The orthographic similarity structure of English words: insights from network science. *Appl. Netw. Sci.* **3**(1), 13 (2018)
44. Stella, M., Brede, M.: Patterns in the English language: phonological networks, percolation and assembly models. *J. Stat. Mech. Theory Exp.* **2015**(5), P05,006 (2015)
45. Turnbull, R., Peperkamp, S.: What governs a language’s lexicon? Determining the organizing principles of phonological neighbourhood networks. In: International Workshop on Complex Networks and Their Applications (2016)
46. Vattani, A., Chakrabarti, D., Gurevich, M.: Preserving personalized PageRank in subgraphs. In: ICML 2011

47. Vitevitch, M.: What can graph theory tell us about word learning and lexical retrieval. *J. Speech Lang. Hear. Res.* **51**(2), 408–422 (2008)
48. Wattenberg, M.: Arc diagrams: visualizing structure in strings. In: INFOVIS 2002
49. Watts, D., Strogatz, S.: Collective dynamics of ‘small-world’ networks. *Nature* **393**(6684), 440 (1998)
50. Yates, M.: How the clustering of phonological neighbors affects visual word recognition. *J. Exp. Psychol. Learn. Mem. Cogn.* **39**(5), 1649–1656 (2013)
51. Zachary, W.: An information flow model for conflict and fission in small groups. *J. Anthropol. Res.* **33**(4), 452–473 (1977)