Simplicial closure and higher-order link prediction

Austin R. Benson
Cornell University
arb@cs.cornell.edu

Rediet Abebe
Cornell University
red@cs.cornell.edu

Michael T. Schaub
MIT and University of Oxford
mschaub@mit.edu

Ali Jadbabaie
MIT
jadbabai@mit.edu

Jon Kleinberg
Cornell University
kleinber@cs.cornell.edu

Abstract

Networks provide a powerful formalism for modeling complex systems, by representing the underlying set of pairwise interactions. But much of the structure within these systems involves interactions that take place among more than two nodes at once — for example, communication within a group rather than person-to-person, collaboration among a team rather than a pair of co-authors, or biological interaction between a set of molecules rather than just two. We refer to these type of simultaneous interactions on sets of more than two nodes as *higher-order interactions*; they are ubiquitous, but the empirical study of them has lacked a general framework for evaluating higher-order models. Here we introduce such a framework, based on *link prediction*, a fundamental problem in network analysis. The traditional link prediction problem seeks to predict the appearance of new links in a network, and here we adapt it to predict which (larger) sets of elements will have future interactions. We study the temporal evolution of 19 datasets from a variety of domains, and use our higher-order formulation of link prediction to assess the types of structural features that are most predictive of new multi-way interactions. Among our results, we find that different domains vary considerably in their distribution of higher-order structural parameters, and that the higher-order link prediction problem exhibits some fundamental differences from traditional pairwise link prediction, with a greater role for local rather than long-range information in predicting the appearance of new interactions.

1 INTRODUCTION

Networks are a fundamental abstraction for complex systems and relational data throughout the sciences [2, 17, 43]. The basic premise of network models is to represent the elements of the underlying system as nodes, and to use the links of the network to capture pairwise relationships — in this way, a social network can represent the friendships between pairs of people; a Web graph can encode links among Web pages or topic categories; and a biological network can represent the interactions among pairs of biological molecules or components [12, 16, 22, 43]. But much of the structure in these systems involves *higher-order interactions* on more than two entities at once [8, 23, 38, 44, 63]: people often communicate or interact in social groups, not just in pairs; associative relations among ideas or topics often involve the intersection of multiple concepts; and joint protein interactions in biological networks are associated with important phenomena [41].

These types of higher-order interactions are apparent even in the standard genres of datasets used for network analysis; for example, co-authorship networks are built from data in which larger groups write papers together; similarly, email networks are based on messages that often have multiple recipients. While higher-order structure is not captured by a graph, it may be modeled via a collection of formalisms that include set systems [21], hypergraphs [9], simplicial complexes [24], and bipartite affiliation graphs [18, 44].

Despite the existence of mathematical formalisms for higher-order structure, it has been challenging to adapt the empirical methodology developed for graph and network data to the higher-order case, due to the lack of general frameworks for evaluating models of higher-order structure. Here we propose such a framework, drawing on the concept of *link prediction*, a fundamental problem in network analysis [34, 36].

Link prediction is a means of evaluating network models by taking network data that evolves over time [7, 26, 32] and seeing how well a given model predicts the appearance of new links — for example, new co-authorships appearing in a co-author network, or new messages between pairs of people in an email network. Link prediction has proved valuable both for methodological reasons and also in concrete applications. Methodologically, asking whether one model is significantly better than another at predicting new links provides a data-driven way of assessing the effectiveness of the models. But since link prediction cuts across many disciplines, it also has a range of direct applications, including predicting friendships in social networks [4, 65], inferring new relationships between genes and diseases [39, 66], and suggesting novel connections in the scientific community [34, 62].

In this paper, we introduce an analogue of link prediction for higher-order structure, providing a general framework for evaluating models in any data where this type of structure evolves over time through the appearance of new, higher-order interactions — for example, predicting which sets (rather than just pairs) of authors will write a paper together, or which sets of people will appear as joint recipients on a new email message. We study the temporal evolution of 19 network datasets from a variety of domains, including social networks, online communication, and biomedicine, where each dataset is a collection of time-stamped sets of nodes, which we call *simplices*. The nodes in each given simplex take part in a shared interaction at the given time-stamp (Fig. 1A). For example, in a co-authorship network, a simplex corresponds to the set of authors of a publication.

The basic premise in link prediction — whether pairwise or higher-order — is to use properties of the structure up to some time t to predict the appearance of new interactions after t. For higher-order

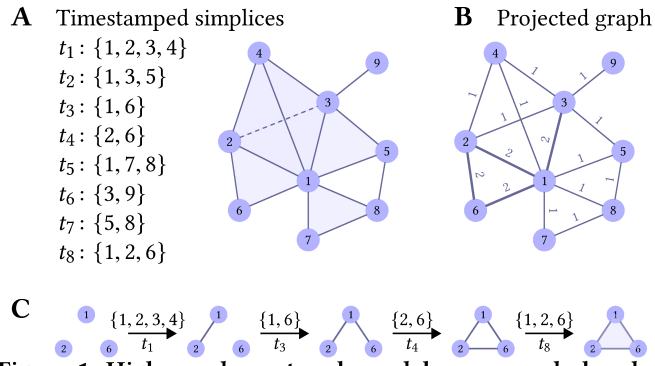


Figure 1: Higher-order network models, open and closed closed triangles, and simplicial closure. (A) Example dataset comprising eight timestamped simplices and nine nodes. The dataset has seven closed triangles—$\{1, 2, 3\}$, $\{1, 2, 4\}$, $\{1, 3, 4\}$, $\{2, 3, 4\}$, $\{1, 3, 5\}$, $\{1, 2, 6\}$, $\{1, 7, 8\}$—and one open triangle—$\{1, 5, 8\}$. (B) The "projected graph" of the dataset. The weight of an edge is the number of times the two end points of the edge appeared in a simplex together. Open and closed triangles are both triangles in the projected graph. (C) Simplicial closure of nodes 1, 2, and 6. Before closing, the three nodes induce several subgraph in the projected graph over time. For example, the nodes form an open triangle at time t_4, which persists until time t_8 when the simplex closes.

structure prediction, which features up to time t are most informative? To develop a set of candidate structural features, we study the *projected graph* of the system, in which two nodes are joined by an edge of weight w if they have been involved in w simplices before time t (Fig. 1B). If there are edges among all pairs from a given set of k nodes — forming a k-clique in the projected graph — this might be because (i) these k nodes were part of a single simplex together, or (ii) because each pair was part of a simplex, although all k were never part of the same simplex. In the former case, we say the k nodes form a *closed* clique, while in the latter case we say they form an *open* clique. Many of the new simplices that form in our data consist of k nodes that had previously constituted an open k-clique in the projected graph; we say that the appearance of the new simplex on these k nodes is an instance of *simplicial closure*, the conversion of an open structure to a closed one (Fig. 1C). Simplicial closure is distinct from the well-known phenomenon of *triadic closure* in social networks [22, 55], since triadic closure modifies the structure of the underlying pairwise interactions, whereas simplicial closure adds a new higher-order interaction without changing the pairwise structure of the projected graph.

We use these pairwise structures in the projected graph, together with higher-order structure, as the basic features in our methods for higher-order link prediction. Among other results, we find the following within our framework.

First, there is enormous diversity across datasets in basic higher-order structural parameters such as the fraction of k-cliques that are open. In particular, for each of our datasets, we look at the edge density in the projected graph and at the fraction of open 3-cliques, and we see that most combinations of values are possible. But beyond this, different datasets from the same domain tend to behave similarly in their edge density and open triangle fraction — for example, low edge density and low fraction of open triangles in co-authorship networks; low edge density and high fraction of open triangles in discussion thread co-occurrence networks; high edge density and low fraction of open triangles in e-mail networks. These findings suggest that there isn't a single "universal" setting of these values; the context underlying the network matters, but within a given context the parameters are quite stable. These results also show the value in comparative analysis of data across a wide range of domains — if we had, say, looked only at co-authorship networks, we might have falsely concluded that the fraction of open triangles could exist only in a narrow range, rather than the much broader range where we find them when looking across multiple domains.

Second, there is an interesting trade-off in the relative predictive power of edge density and edge weight. For the case of $k = 3$, for example, a greater number of edges among the k nodes is predictive of the arrival of a simplex; but higher weight on these edges is also predictive. Thus, it is not a priori clear whether a *light triangle* on 3 nodes (an open 3-clique with all edge weights small) is more or less predictive of a new 3-node simplex than a *heavy wedge*, with high weight on two edges and a missing third edge. The datasets we study differ in which of these two structures has stronger predictive value. However, there is consistency in two respects: the datasets within a single domain show approximately the same relative effectiveness between these two structures; and within a dataset, the relative predictive power of edge density and edge weight carries over one dimension up, when we think about the closure of 4-node simplices.

Finally, the link prediction problem for higher-order structure exhibits some fundamental differences from traditional link prediction with pairwise interactions (i.e. for the edges in a graph). In the traditional link prediction problem, a key principle is that it is valuable to use information contained in paths of non-trivial length between two nodes u and v for predicting a link between them — for example, PageRank-like measures and other forms of path enumeration are effective [34, 36]. But we find that something different happens in higher-order link prediction: to predict the formation of a simplex on nodes u, v, and w, it is difficult to improve on the purely *local* information contained in the three-dimensional vector of edge weights for (u, v), (v, w), and (u, w) in the projected graph. In this sense, higher-order link prediction is a task that seems fundamentally more local in its overall structure than the traditional pairwise link prediction problem. This appears to arise from the ability of a k-tuple of nodes, for $k \geq 3$, to contain rich local information in its interactions among subsets of size $k - 1$ — a phenomenon that has no natural analogue in the case of $k = 2$, and which renders the prediction problems qualitatively more distinct than one might initially suppose.

Despite the power of local information, we still find that methods based on supervised learning, combining multiple structural features, can produce improvements over any one feature in isolation. To enumerate features for this purpose, we employ measures not just based on parameters of the projected graph, but also set-based measures that treat the higher-order structure more directly. The fact that combinations of these features prove effective

Table 1: Summary counts for datasets. Each dataset is a collection of timestamped simplices.

Dataset	nodes	edges in proj. graph	timestamped simplices	unique simplices
coauth-DBLP	1,924,991	7,904,336	3,700,067	2,599,087
coauth-MAG-Geology	1,256,385	512,0762	1,590,335	1,207,390
coauth-MAG-History	1,014,734	1,156,914	1,812,511	895,668
music-rap-genius	56,832	123,889	224,878	85,429
tags-stack-overflow	49,998	4,147,302	14,458,875	5,675,497
tags-math-sx	1,629	91,685	822,059	174,933
tags-ask-ubuntu	3,029	132,703	271,233	151,441
threads-stack-overflow	2,675,955	20,999,838	11,305,343	9,705,709
threads-math-sx	176,445	1,089,307	719,792	595,778
threads-ask-ubuntu	125,602	187,157	192,947	167,001
NDC-substances	5,311	88,268	112,405	10,025
NDC-classes	1,161	6,222	49,724	1,222
DAWN	2,558	122,963	2,272,433	143,523
congress-bills	1,718	424,932	260,851	85,082
congress-committees	863	38,136	679	678
email-Eu	998	29,299	234,760	25,791
email-Enron	143	1,800	10,883	1,542
contact-high-school	327	5,818	172,035	7,937
contact-primary-school	242	8,317	106,879	12,799

in many domains highlights the richness of the underlying problem, and we believe the array of methods presented here can potentially help suggest further progress on these questions in higher-order structure.

2 STATIC ANALYSIS: OPEN AND CLOSED TRIANGLES

Before getting to the higher-order link prediction problem, we first provide a brief overview and analysis of the basic structure of our datasets. We represent each dataset by a collection of N timestamped simplices, $\{(S_i, t_i)\}_{i=1}^{N}$, where $t_i \in \mathbb{R}$ is the time at which simplex S_i was observed in the data. Each simplex S_i is a subset representing the nodes in the ith simplex. If $|S_i| = k$, we say that S_i is a k-node simplex.[1] This set-based representation provides a natural format for datasets from a range of domains. For the present study, we focus on the following large collection of 19 datasets:

- *Co-authorship data* (coauth-DBLP, coauth-MAG-History, coauth-MAG-Geology): nodes are authors and a simplex is a publication.
- *Online tagging data* (tags-stack-overflow, tags-math-sx, tags-ask-ubuntu): nodes are tags (annotations) and a simplex is a set of tags for a question on a question-and-answer forum.
- *Online thread participation data* (threads-stack-overflow, threads-math-sx, threads-ask-ubuntu): nodes are users and a simplex is a set of users answering a particular question on a question-and-answer forum.
- *Drug networks from the National Drug Code Directory* (NDC-classes, NDC-substances): in the first dataset, nodes are class labels (e.g., serotonin reuptake inhibitor) and a simplex is the set of class labels applied to a drug (NDC-classes); in the second dataset, nodes are substances (e.g., testosterone) and a simplex is the set of substances in a drug.
- *U.S. Congress data* (congress-committees [54], congress-bills [20]): nodes are members of Congress and a simplex is the set of members in a committee or co-sponsoring a bill.
- *Email networks* (email-Enron [29], email-Eu [48]): nodes are email addresses and a simplex is the set of addresses sending or receiving an email.
- *Contact networks* (contact-high-school [37], contact-primary-school [61]): nodes are persons and a simplex is a set of persons in close proximity to one another at a given time.
- *Drug use from the Drug Abuse Warning Network* (DAWN): nodes are drugs and a simplex is the set of drugs reportedly used by a patient prior to an emergency department visit.
- *Music collaboration* (music-rap-genius): nodes are rap artists and a simplex is a set of artists collaborating on a song.

All datasets except the U.S. congress committee membership and music collaboration datasets are available at

`http://www.cs.cornell.edu/~arb/data/`.

Summary statistics of the datasets are in Table 1, and Appendix A provides more details on the datasets. To provide uniformity across datasets, we restrict to simplices consisting of at most 25 nodes. This is relevant to, e.g., the co-authorship data in which large consortia of hundred or more authors may collaborate on a single paper. However, such events are rare and are not relevant for our discussion below.

We focus in the following on the distinction between open versus closed triangles, as this corresponds to the simplest case of simplicial closure as defined above. Furthermore, triangles are one of the most important structural patterns in network analysis [30, 33, 38, 56]. As discussed above, there are two types of triangles (Fig. 1), which cannot be distinguished by the weighted projected graph alone. In a *closed triangle*, all three nodes have co-appeared in at least one simplex. Formally, $\{u, v, w\}$ is a closed triangle if there exists some simplex S_i for which $\{u, v, w\} \subset S_i$. In an *open triangle*, on the other hand, every pair of the three nodes has co-appeared in at least one simplex, but no single simplex contains all three nodes.

Every simplex with at least 3 nodes will directly create a closed triangle, while open triangles are essentially coincidental. Thus, one might intuit that closed triangles should be much more common than open triangles in our data. Moreover, larger simplices lead to many closed triangles: for instance, a k-node simplex contributes $\binom{k}{3}$ closed triangles. Surprisingly, however, our analysis reveals that open triangles are more common than closed ones in most of our datasets (Figs. 2C and 2D). While the distribution of simplex sizes is broadly similar in most datasets (Fig. 2B), jointly analyzing the fraction of triangles that are open with the edge density in the projected graph reveals a rich landscape of datasets (Fig. 2C): (i) low-density with a small fraction of open triangles (co-authorships and music collaboration); (ii) low-density with a large fraction of open triangles (stack exchange threads) (iii) high-density with a large fraction of open triangles (stack exchange tags, contact networks, Congress bill co-sponsorship); and (iv) high-density with medium fraction of open triangles (email, Congress committee membership, NDC substances and classes). Remarkably, these results are not skewed by large simplices — the picture is broadly preserved when restricting the datasets to only the 3-node simplices (Fig. 2E).

[1] Such a structure would be called a (k-1)-simplex in algebraic topology, and the set of all its pairs would be called a k-clique in graph theory.

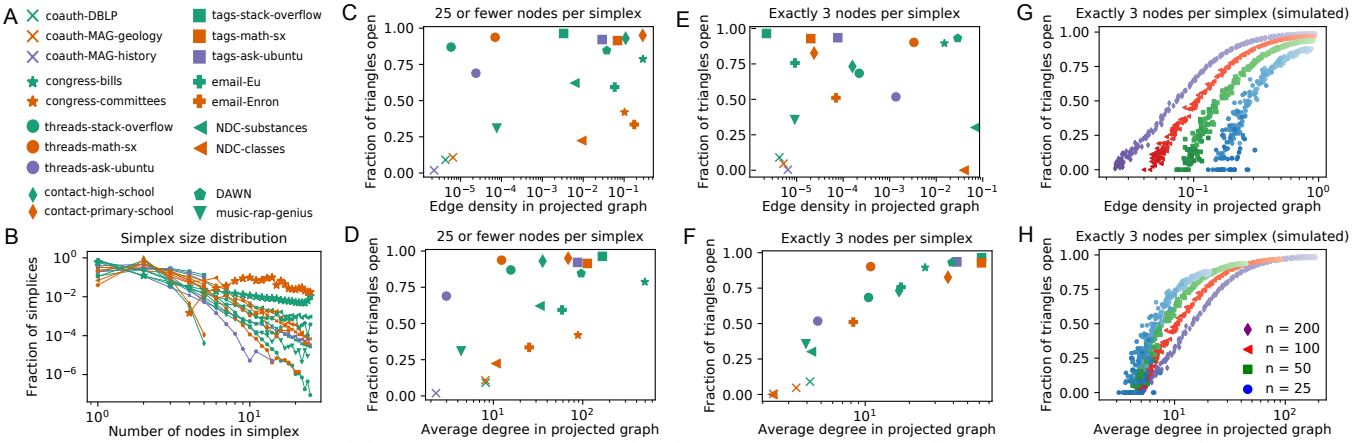


Figure 2: Structure of datasets. (A) Legend for subfigures (B–F). Within each domain (denoted by marker type), datasets are ordered by total number of simplices (green being the largest, orange the second largest, and purple the smallest number simplices). (B) Distribution of simplex sizes. In most datasets, small simplices (≤ 4 nodes) are the most common. The congress-committees dataset has no 3-node simplices, so it does not appear in subfigures (E–F). (C–F) Dataset landscapes in terms of fraction of triangles that are open and either edge density (C,E) or average degree (D,F) when considering simplices with 25 or fewer nodes (C–D) or just 3-node simplices (E–F). Datasets from the same domain tend to be similar with respect to these features, whether or not we include simplices with greater than 3 nodes. (G–H) Distribution of fraction of open triangles and either edge density (G) or average degree (H) in simulated data from a model where each triple of n total nodes forms a 3-node simplex with probability $p = 1/n^b$, $b \in [0.8, 1.8]$. Color scales with b so that smaller larger p are lighter and smaller p are darker. Varying b creates datasets spanning many fractions of open triangles.

If we measure average unweighted degree instead of edge density (along with fraction of open triangles) we again find a substantial diversity, and datasets from the same domain behave similarly with respect to these two features (Fig. 2D). Restricting the data to only 3-node simplices, we find a near-linear relationship between the fraction of open triangles and the log of the average degree (Fig. 2F). A linear model for the data in Fig. 2F has standard errors of 0.08 for the intercept and 0.03 for the log average degree regression coefficient with $R^2 = 0.85$, compared to standard errors of 0.16 and 0.04 with $R^2 = 0.38$ for a linear model for the data in Fig. 2D. This suggests that larger simplices lead to diversity in the data.

So why is there an abundance of open triangles, which seems to contradict intuition? One extreme hypothesis is that 3-node simplices form independently with a fixed probability. In this case, open triangles can indeed become much more common than closed ones. To see this, suppose that a dataset consists only of 3-node simplices, and a given set of three nodes $\{u, v, w\}, 1 \leq u, v, w \leq n$ is a simplex with probability $p = 1/n^b$, for $b > 0$. Let X_{uvw} be the indicator random variable that $\{u, v, w\}$ is an open triangle. Then, for large n, it follows from the independence assumption that

$$\mathbb{E}[X_{uvw}] \approx (1 - (1 - 1/n^b)^n)^3. \tag{1}$$

There are two asymptotic regimes here depending on the value of b. If $b < 1$, then $(1 - 1/n^b)^n \leq e^{-n^{1-b}}$, and $\mathbb{E}[X_{uvw}]$ approaches 1 as n gets large. If $b > 1$, on the other hand,

$$\mathbb{E}[X_{uvw}] \approx (1 - (1 - 1/n^b)^n)^3 = O(1/n^{3b-3}), \tag{2}$$

Let us now denote the set of open triangles by O and the set of closed triangles by C. According to our calculations above, for large n, the expected number of open triangles would be $\mathbb{E}[|O|] = \sum_{\{u,v,w\}} \mathbb{E}[X_{uvw}] = O(n^3)$ if $b < 1$. For $b > 1$ the expected number of open triangles for large n is $\mathbb{E}[|O|] = O(n^{3(2-b)})$. In contrast to the case of open triangles, the expected number of closed triangles is always $\mathbb{E}[|C|] = p \cdot \binom{n}{3} = O(n^{3-b})$. Therefore, if $b < 3/2$, the number of open triangles grows faster, whereas if $b > 3/2$, the number of closed triangles grows faster. To summarize, if the probability of 3-node simplex to form is large ($p > 1/n^{3/2}$) and there are many nodes (n is large), then we can expect more open triangles than closed ones under this simple independent model.

To illustrate these points numerically, we generate 5 random samples from this model for $b = 0.8, 0.82, 0.84, \ldots, 1.8$ and $n = 25, 50, 100, 200$. As suggested by the above calculations, the samples drawn have a fraction of open triangles spanning the interval between 0 and 1 (Figs. 2G and 2H). When the number of nodes n is larger, the edge density is smaller (Fig. 2G) and the average degree is larger (Fig. 2H).

We can also use the above procedure to form datasets with a smaller edge density, while keeping the average degree fixed by simply patch together c replicates of one of these random datasets, creating a dataset with c times as many nodes, but the same average degree. More formally, if a dataset with n nodes has average degree d and edge density ρ, then the union of c copies of this dataset has cn nodes, average degree d, and edge density $c\rho(\binom{n}{2} - n)/(\binom{nc}{2} - nc) \approx \rho/c$ (for large n). Thus, our simple independent model spans the two-dimensional feature space in Figs. 2C and 2E.

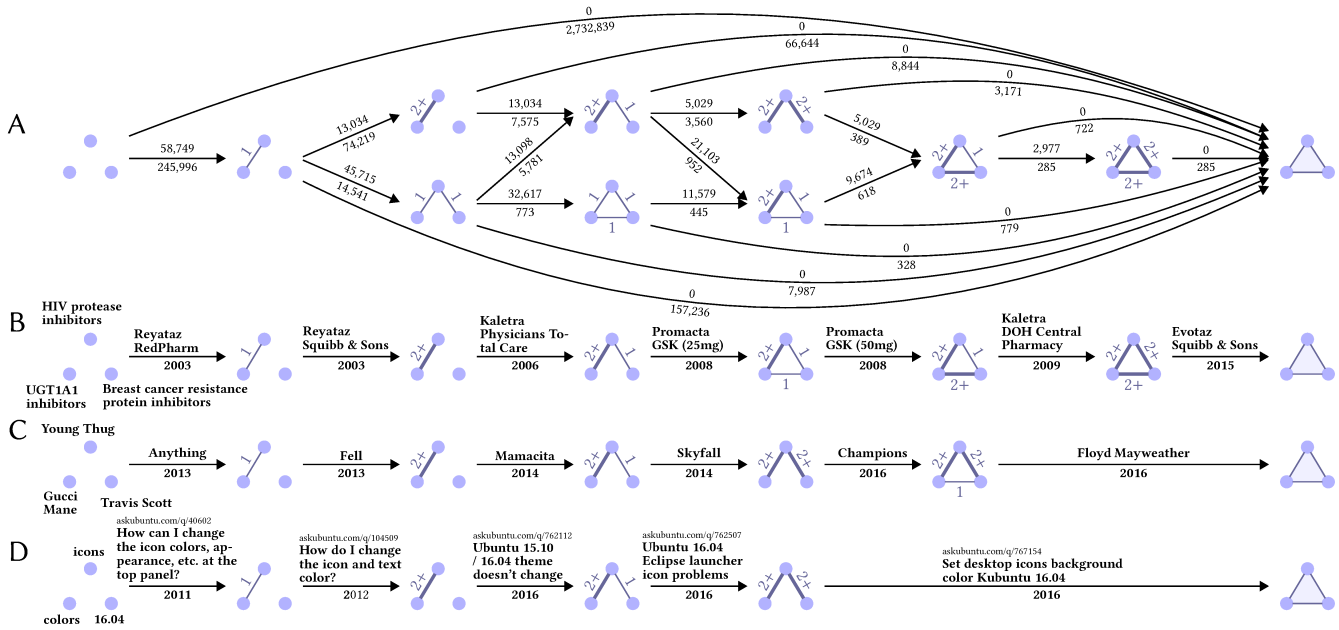


Figure 3: Lifecycles of triples of nodes for. Triangle edge weights are edge weights in the projected network binned into weak ties for pairs of nodes appearing in only one simplex together (denoted "1") and strong ties for pairs of nodes appearing at least two simplices together (denoted "2+"). (A) Lifecycles of open and closed triangles in the coauth-MAG-History dataset. Edges represent transitions between weighted induced subgraphs in the project network, and the numbers are counts of triples that go through the transition. The top number counts triples of nodes that never simplicially close (i.e., never reach the closed state on the far right), and the bottom number counts triples that simplicially close. (B) Lifecycle of classification codes "HIV protease inhibitors", "UGT1A1 inhibitors", and "Breast cancer resistance protein inhibitors" in the NDC-classes dataset, where simplices consist of the labels applied to drugs. Reyataz and Kaletra—two HIV-1 medications—produced strong ties via multiple drug labelers; RedPharm Drug Inc. and E.R. Squibb & Sons, LLC labelled Reyataz, and Physicians Total Care and DOH Central Pharmacy labelled Kaletra. Promacta, a bone marrow stimulant classified as both a breast cancer resistance protein inhibitor and a UGT1A1 inhibitor, creates the open triangle. A strong tie is due to GlaxoSmithKline plc labelling multiple dosages of Promacta as products (25mg and 50mg). Evotaz, a combination drug, provided simplicial closure for the three labels, 6 years after the open triangle formed. (C) Lifecycle of rap artists Young Thug, Gucci Mane, and Travis Scott. Mane and Thug first collaborated on the song "Anything" on a Mane mixtape; the two subsequently both featured on Waka Flocka Flame's track "Fell". Thug then twice featured on Travis Scott's 2014 mixtape "Days Before Rodeo", on the tracks "Mamacita" and "Skyfall". Both Mane and Scott featured on Kanye West's ensemble track "Champions" leading to an open triangle. Simplicial closure occurred when Scott and Mane both featured on Thug's track "Floyd Mayweather". (D) Lifecycle of tags "icons", "colors", and "16.04" applied to questions on the Ask Ubuntu question-and-answer forum. The tag 16.04 refers to a 2016 Ubuntu release. There are questions about icons and colors independent of the Ubuntu version, dating back to 2011 (just 1 year after the web site started). The 16.04 tag comes to use in 2016, at which point a couple 16.04-specific icon questions are asked. Finally, a 16.04-specific question on both icons and colors leads to simplicial closure.

3 HIGHER-ORDER TEMPORAL EVOLUTION: SIMPLICIAL CLOSURE

While the static analysis above already contains interesting information about the organization of closed and open triangles, the temporal dynamics of the networks offer additional insights. One plausible explanation for open triangles might be temporal asynchronicity. For example, with regards to the Congress committee membership dataset, consider three congresspersons u, v, and w, where, in one congress, u is in one committee with v and another with w. If u is not re-elected, then there will be no opportunity for the triple of nodes to form a closed triangle, as u has effectively become inactive. An open triangle may still form if v and w are on the same committee in a future Congress. However, we find that temporal asynchronicity does not explain most open triangles. Depending on the dataset, the three edges in 61.1% to 97.4% of open triangles have an overlapping period of activity (including 89.5% for Congress committees; Table 3).

Regardless of the open triangle creation processes, the three nodes in an open triangle may form a simplex in the future as the network evolves, a process ignored by existing graph-based (dyadic) models. In the following, we study *simplicial closure*, the

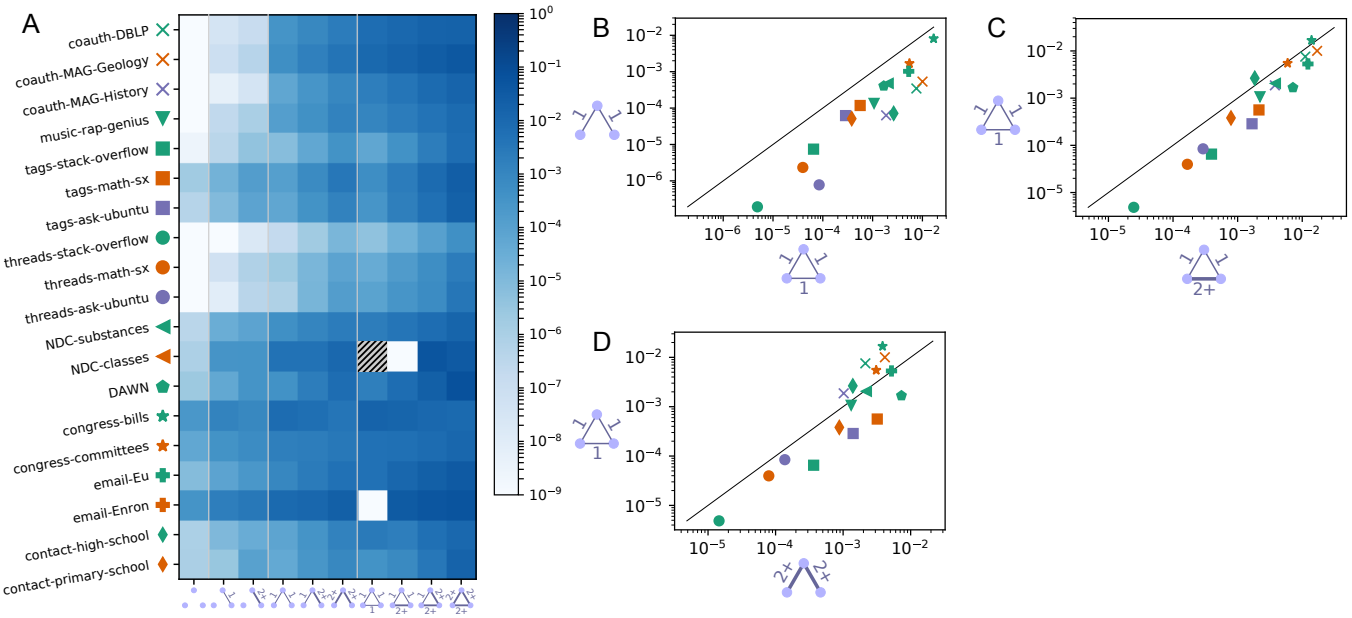


Figure 4: Closure probability as a function of the 3-node weighted induced subgraph configuration in the projected graph. The simplices appearing in the first 80% of the time spanned by the dataset determine the configuration of every triple of nodes in the projected graph. We measure the probability that 3 nodes appear together in a simplex in the final 20% of timestamped simplices, conditioned on the prior subgraph configuration. (A) Heat map of simplicial closure probabilities for every 3-node weighted induced subgraph configuration. The shaded box is the only case with fewer than 20 samples. The four sections of the heat map correspond to 0, 1, 2, or 3 edges in the induced subgraph. (B–C) Comparison of closure probabilities for pairs of 3-node configurations from (A) that demonstrate how increasing edge density (B) and tie strength (C) increase simplicial closure probability. (D) Neither edge density nor tie strength dominates the influence on simplicial closure. Depending on the dataset, open triangles of all weak ties (the "weak open triangle") or just two strong ties (the "strong wedge") are more likely to close.

process by which nodes form simplices over time. We first examine triangles, the simplest higher-order structure and then study larger structures in Section 3.2.

3.1 Simplicial closure on three nodes

Any induced subgraph on three nodes in the weighted projected graph can change several times before the three nodes appear in a simplex together (Fig. 3), i.e., simplicially close. We call this the *lifecycle* of the triplet of nodes. There are two changes that a triplet of nodes can undergo during its lifecycle before simplicial closure. First, an extra edge can be added between two nodes u and v. This corresponds to an increase in density in this induced subgraph, e.g., the introduction of the drug Promacta adds an edge in Fig. 3B. Second, projected graph edge weights can increase; we interpret this as increasing tie strengths. For instance, in Fig. 3C, the tie strength between Gucci Mane and Young Thug increases after they collaborate on "Fell". To simplify our analysis, we differentiate only between *weak ties* with weight $W_{uv} = 1$ and *strong ties* corresponding to multiple interactions over time ($W_{uv} \geq 2$). The ties are denoted by "1" (weak) and "2+" (strong) in Fig. 3, and there are 11 possible states in a lifecycle (Fig. 3A).

Our prediction tasks will be focused on closure probabilities. However, to get an impression of the absolute magnitude of those numbers, we examine the lifecycle of every triple of nodes that becomes an open or closed triangle in the coauth-MAG-History dataset (Fig. 3A). In this dataset, a closed triangle is more likely to have come from a configuration with exactly two strong ties edges (3,171 cases) than from an open triangle (328 + 779 + 722 + 285 = 2,114 cases). Most closed triangles are formed by nodes that had no previous interaction (2,732,839 cases), but there are also many such triples of nodes. We also see that if three nodes form an open triangle with only weak ties at some point in time, then the three nodes are much more likely to gain a strong tie before closure (445 cases) than to close directly from that state (328 cases).

We also compute the probability of simplicial closure conditioned on the state of the three nodes in its lifecycle. We split each of our datasets based on the temporal order of appearance of the simplices into a training set, consisting of the first 80% of the simplices (in time) and a test set of the remaining 20% of the simplices. Formally, if t_* is the 80th percentile of the timestamps $t_1, \ldots, t_N$, then the training set is the set of timestamped simplices $\{(S_i, t_i) \mid t_i \leq t_*\}$ and the test set consists of $\{(S_i, t_i) \mid t_i > t_*\}$. (Our results are consistent if we predict at different points of time; see Appendix C.) We then measured the probability that a node-triple from the training set will form a closed triangle in the test set as a function of its previous configuration in the weighted projected graph, i.e., its lifecycle state in the training data. This gives 10 closure probabilities for each dataset—one for each of the 10 open configurations

in Fig. 3. Figure 4 summarizes the dependence of simplicial closure probability on the node set configuration.

We highlight a few important findings. First, simplicial closure probability typically increases with additional edges (Fig. 4B). In other words, as the edge density of the subgraph induced by the three nodes increases, the simplicial closure probability increases. We formally test this by comparing the closure probability of a fixed weighted induced subgraph configuration and the same configuration with an additional unit-weight edge for all suitable cases. The latter has a statistically significant higher closure probability in 102 of 113 cases over all datasets and pairs of configurations, whereas the less dense structure is never significantly more likely to close ($p < 10^{-5}$; see Section 3.3 for a more detailed description on the hypothesis tests). (Our goal here is to illustrate general trends rather than to find a single statistically significant result.) This result is consistent with both the theory of social networks [22] and empirical studies of social networks [32] on dyadic link formation. However, many of our datasets are not social networks.

Second, the simplicial closure probability typically increases with tie strength (Fig. 4C). We test the effect of tie strength by comparing the closure probability of a fixed weighted induced subgraph containing at least one weak tie, and the same configuration where the weak tie is converted to a strong tie. Increasing the tie strength significantly increases the closure probability in 82 of 113 cases over all datasets and significantly decreases the closure probability in just 6 of 113 cases ($p < 10^{-5}$). Again, these results are consistent with both theory [22] and empirical studies of social networks [3, 30].

However, neither edge density nor tie strength dominates the influence on simplicial closure (Fig. 4D). In the co-authorship and Congress datasets, an open triangle comprised of three weak ties, which we call the *weak open triangle* is more likely to close than a 3-node subgraph with just two strong ties, which we call the *strong wedge*. The reverse is true for the stack exchange tags and stack exchange threads datasets. Overall, a weak open triangle is statistically significantly more likely to close than the strong wedge in 4 of 19 datasets, whereas the strong wedge is significantly more likely to close in 6 of 19 datasets ($p < 10^{-5}$). These results suggests different closure dynamics for the different dataset domains. In human social interactions, simplicial closure appears to be driven by a topological form of triadic closure: mutual acquaintance between all the nodes in a set increases the probability of a joint interaction. In contrast, simplicial closure in the discussion platform networks resemble transitive closure: once there is a sufficiently strong co-occurrences of tags, they are likely to be used together later on.

3.2 Simplicial closure on four nodes

We now study simplicial closure on four nodes and again measure the probability of closure as a function of the 4-node configuration in the training data. In the case of 3-node simplicial closure, we determined the open configuration in the training set by examining the three nodes in the the projected graph. Effectively, we examined how many times each of the three 2-*node subsets* co-appeared in a simplex. For 4-node open configurations we proceed analogously, using the corresponding 3-*node subsets*. Specifically, for a given set of 4 nodes, every triangle in the projected graph is classified as either (i) an open simplicial tie, i.e., the triangle is open; (ii) a weak simplicial tie, meaning that the 3 nodes have appeared in just one simplex together; or (iii) a strong simplicial tie, meaning that the three nodes have appeared in at least two simplices together. In contrast to the 3-node case, these 4-node configurations are not completely determined by the weighted projected graph, since the projected graph (as defined) does not contain information on whether or not 3 nodes induce a closed triangle. Thus, with 4-node simplices, we make use of additional topological information provided in our set-valued datasets.

One could also account for the tie strengths of the 2-node subsets (edges) in a 4-node configuration — a complete characterization of the possible induced configurations leading to simplicial closure is much more complex for the 4-node case than the 3-node case. For an accessible study on the closure patterns of 4-node simplices, we measure the probability of closure with respect to the 27 open configurations where the induced 4-node subgraph in the projected graph contains at least one triangle. Our classification distinguishes triangles by tie strength (open, weak, or closed), but not by edge tie strengths, other than by what is implied by the triangles.

Analogous to the experiments in Section 3.1, we measure the probability that a subset of four nodes closes in the test data, conditioned on the open configuration of the same nodes in training data (Fig. 5A; see Appendix D for efficient algorithms for computing these probabilities).

We again find that edge density and tie strength increase the likelihood of simplicial closure. To measure the effect of density, we compare the closure probability of a configuration consisting of a fixed number of edges to the closure probability of the same configuration with an additional edge, keeping the tie strengths of the triangles fixed (Fig. 5B shows one such comparison). In 180 of 228 applicable comparisons over all datasets, the closure probability significantly increases with the edge density, and in only 2 cases, the closure probability decreased significantly ($p < 10^{-5}$).

To measure the effect of simplicial tie strength, we compare the closure probability of a given configuration to the closure probability of the same configuration where the simplicial tie strength of a triangle increases from an open tie to weak tie or from a weak tie to a strong tie (Fig. 5C shows a case where the strength of a triangle changes from open to weak). The closure probability significantly increases with simplicial tie strength in 26 of 38 cases for 3-edge configurations, 31 of 38 cases for 4-edge configurations, 77 of 114 cases for 5-edge configurations, and 177 of 359 cases for 6-edge configurations; compared to a significant decrease in closure probability in just 2 of 38, 1 of 38, 1 of 114, and 4 of 359 cases ($p < 10^{-5}$). Therefore, our finding that tie strength is a positive indicator of simplicial closure between three nodes extends to the case of four nodes.

Similar to the case of three nodes, there is an ambiguity about the influence of sparser configurations comprising strong ties and denser configurations of weak ties. In some datasets, the strong ties are more indicative of simplicial closure, despite possible sparsity of the configuration compared to the closed 4 node simplex; in other datasets, the density is more indicative. To illustrate this tension, we compare the closure probability of the open configuration

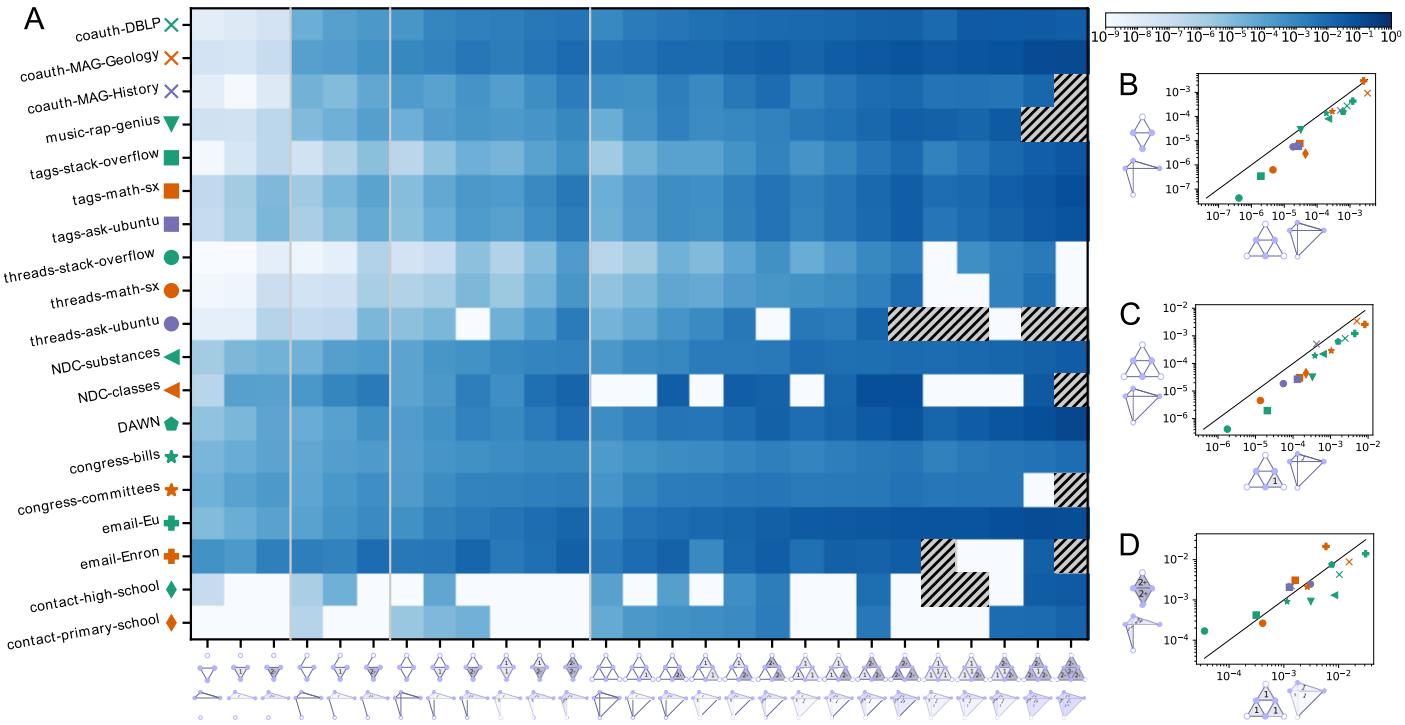


Figure 5: Simplicial closure probability as a function of the 4-node open configuration. We use the first 80% of the timestamped data to determine the configuration of every 4-node set that contains at least 1 triangle and does not appear in a simplex. We then compute the probability that a 4-node set appears in a simplex in the final 20% of the data, conditioned on the open configuration. In an open configuration, there are three types of simplicial tie strengths for a triangle—open, weak, and strong—given by the number of times the three nodes in the triangle have co-appeared in a simplex (zero, one, or at least two times). (A) Heat map of simplicial closure probabilities as a function of open 4-node configuration. Shaded boxes are configurations that appear 20 or fewer times in the first 80% of the data. We illustrate each subgraph configuration on the x-axis with a projection of the simplex onto two dimensions (top line—the unfilled circle represents the same node) as well as a tetrahedral three-dimensional perspective figure (bottom line). The four sections of the heat map correspond to 3, 4, 5, or 6 edges in the configuration. (B–C) Comparison of closure probabilities for pairs of 4-node configurations that demonstrate how increasing edge density (B) and simplicial tie strength (C) lead to higher probability of simplicial closure. (D) Similar to the case of open 3-node configurations (Fig. 4D), in some datasets, open configurations with a smaller number of strong simplicial ties are more likely to close than open configurations with a larger number of weak simplicial ties; in other datasets, the opposite is true. Here, we compare the "strong flap" configuration (y-axis) to the "weak wireframe" configuration (x-axis).

with 5 edges and 2 strong simplicial ties, which we call the *strong flap*, to the open configuration with 6 edges and 3 weak simplicial ties, which we call the *weak wireframe* (Fig. 5D). The closure probability of the strong flap is significantly more likely to close in 4 of 19 datasets, whereas the weak wireframe is significantly more likely to close in 5 of 19 datasets. In the remaining 10 datasets, neither closure probability was significantly higher ($p < 10^{-5}$).

Our observations about the relative importance of tie strength versus tie density (relative to the assesed simplex), translates across the dimensions of the simplices. In three of the five datasets for which the wireframe is significantly more likely to close, the weak open triangle is also significantly more likely to close (coauth-DBLP, coauth-MAG-Geology, congress-bills). And in three of the four datasets for which the strong flap is significantly more more likely than the weak wireframe configuration to close, the strong wedge is also more likely to close than the weak open triangle (tags-stack-overflow, tags-math-sx, tags-ask-ubuntu). Moreover, there were no datasets for which tie strength was significantly more indicative of simplicial closure for one simplex size and density was more important for another. This provides additional evidence for simplicial closure governing mechanisms, as discussed in Section 3.1 — closure is more "topological" in human social networks and more "transitive" in discussion platform tags.

3.3 Details on hypothesis tests

Let us assign labels $c = 1, 2, \ldots, 10$ to the open 3-node configurations, corresponding to the 10 configurations from left to right along the x-axis of Fig. 4A (see Table 6). We denote by n_c the instances of an open configuration c in the training data, and use x_c to denote the number of those instance that close in the test set. For a pair of configurations c_1 and c_2, we can use a one-sided

hypothesis test for $x_{c_1}/n_{c_1} < x_{c_2}/n_{c_2}$. We use Fisher's exact test when the maximum of x_{c_1} and x_{c_2} is less than 5; otherwise, we use a one-sample z-test. The edge density tests compared configuration pairs c_1-c_2 equal to 1-2, 2-4, 3-5, 4-7, 5-8, and 6-9. The tie strength tests compared configuration pairs 2-3, 4-5, 5-6, 7-8, 8-9, and 9-10.

Similarly, we label the open 4-node configurations $c = 1, 2, \ldots, 27$, corresponding to the 27 configurations from left to right along the x-axis of Fig. 5A (again, see Table 6). The edge density tests compared configuration pairs 1-4, 2-5, 3-6, 4-7, 5-8, 6-9, 7-13, 8-14, 9-15, 10-16, 11-17, and 12-18. The tie strength tests compared configuration pairs 1-2 and 2-3 for 3-edge configurations; 4-5 and 5-6 for 4-edge configurations; 7-8, 8-9, 10-11, 11-12, 8-10, and 9-11 for 5-edge configurations; and 13-14, 14-15, 16-17, 17-18, 19-20, 20-21, 21-22, 23-24, 24-25, 25-26, 26-27, 14-16, 15-17, 16-19, 17-20, 18-21, 19-23, 20-24, 21-25, and 22-26 for 6-edge configurations.

4 EVALUATION FOR HIGHER-ORDER LINK PREDICTION

Up to this point, we have studied simplicial closure in aggregate and found that edge density and tie strength are significant positive indicators for simplicial closure. We now evaluate algorithms for higher-order link prediction that predict candidate sets of nodes for simplicial closure. For simplicity of presentation and scalability reasons, we focus on predicting simplicial closure of node-triplets. Thus, the higher-order link prediction problem considered here is predicting which node-triplets that have not co-appeared in a simplex together yet will be a subset of a single simplex in the future.

Our results in Section 3.1 suggest that open triangles or node-triplets with strong ties are the most likely to close in the future. For our experiments, we focus on predicting which open triangles in the training data simplicially close in the test data. By focusing on the prediction of open triangles, it is feasible to enumerate all open structures upon which the algorithms will make a prediction, using only modest computational resources. Thus, we avoid a common problem in link prediction of how to pare down an enormous candidate set of potential links, which itself is an active research topic [6, 58].

4.1 Algorithms

Inspired by classical work on link prediction, we develop several algorithms that assign a score to every open triangle; sorting by these scores induces a ranking of the open triangles corresponding to the algorithm's confidence that each given open triangle will close. More formally, each algorithm defines a score function $s\colon V \times V \times V \to \mathbb{R}$, where $s(i,j,k) > s(a,b,c)$ means that the candidate node-triplet $\{i,j,k\}$ is predicted to be more likely to co-appear in a simplex in the future than $\{a,b,c\}$. Most of the score functions are valid for any triple of nodes, so it is convenient to write s as a function on triples of vertices, rather than just on open triangles. The ordering of the triples induced by the score function will be sufficient for our evaluation in Section 4.2.

We consider score functions in four broad categories (see Appendix E for a complete description). In the first category, $s(i,j,k)$ depends only on the weights of the edges of the triangle in the projected graph. Specifically, we consider the harmonic, geometric, and arithmetic means of the edge weights, which has the property that stronger ties lead to larger scores. These score functions capture our finding that tie strength is a positive indicator of simplicial closure.

The second category of score functions is based on local neighborhood features in the projected graph such as the common neighbors of nodes i, j, and k. Three score functions in this category use the number of common fourth neighbors of the 3 nodes in the projected graph with (possible) normalization given by generalizations of the Jaccard similarity or Adamic-Adar similarity [1, 34]. These methods are direct generalizations of score functions used for dyadic link prediction [34]. We also evaluate two score functions inspired by preferential attachment, which has been suggested as a growth mechanism of co-authorship networks [7, 42]. These score functions are given by the product of the number of neighbors in the projected graph of the three nodes or the product of the number of simplices in which each of the three nodes appear.

The third category of score functions is based on paths and random walks, a successful methodology for dyadic link prediction [34]. We compute pairwise similarities between nodes for weighted and unweighted Katz similarity and personalized PageRank similarity. The score is then the sum of the pairwise scores between the three nodes. We also use a recent generalization of personalized PageRank that accounts for higher-order structure using ideas from computational topology [27], which we call *simplicial PageRank*. This method computes pairwise similarity scores between edges, and the score function is the sum of the pairwise scores between the edges in an open triangle. This method is also the most computationally expensive and did not complete for two of the datasets within 2 weeks of computation time. The similarity scores can be further decomposed into three components based on the Hodge decomposition [35], which can have a substantial effect on the prediction performance (see Appendix E).

The final category of score function uses a supervised learning approach by automatically learning the importance of several features derived from the other score functions. More specifically, we train a regularized logistic regression model using 26 features of the triple of nodes (see Appendix E), and the score function is the probability of simplicial closure given by the model on the test set. While there are many methods for feature-based supervised learning, we found logistic regression to work well in practice, and we show that the supervised approach is competitive with (and often out-performs) the methods described above.

4.2 Prediction performance

Using the ranking induced by the score functions described above, we evaluated the prediction performance on each dataset by the area under the precision-recall curve (AUC-PR) metric (Table 2). AUC-PR is particularly appropriate for prediction problems with class imbalance [15], which is the case for our datasets (Fig. 4). We use random scores — more specifically, a random ranking — as a baseline. The baseline prediction performance with respect to AUC-PR is the proportion of open triangles in the training set that simplicially close in the test set.

Table 2: Open triangle closure prediction performance based on several score functions: random (Rand.); harmonic, geometric, and arithmetic means of the 3 edge weights (Eqs. (19) to (21)); 3-way common neighbors (Common, Eq. (22)); 3-way Jaccard coefficient (Jaccard, Eq. (23)); 3-way Adamic-Adar (A-A, Eq. (24)); projected graph degree and simplicial degree preferential attachment (PGD-PA, Eq. (25) and SD-PA, Eq. (25)); unweighted and weighted Katz similarity (Katz, Eq. (29) and W-Katz, Eq. (30)); unweighted and weighted personalized PageRank (U-PPR, Eq. (34) and W-PPR, Eq. (35)); simplicial personalized PageRank (S-PPR, Eq. (42); the two missing entries are cases where computations did not finish within 2 weeks); and a feature-based supervised method logistic regression (Log. reg.). Performance is AUC-PR relative to the random baseline. The random baseline is listed in absolute terms and equals the fraction of open triangles that close.

Dataset	Rand.	Harm. mean	Geom. mean	Arith. mean	Common	Jaccard	A-A	PGD-PA	SD-PA	U-Katz	W-Katz	U-PPR	W-PPR	S-PPR	Log. reg.
coauth-DBLP	1.68e-03	1.49	1.59	1.50	1.33	1.84	1.60	0.74	0.74	0.97	1.51	1.62	1.83	1.21	3.37
coauth-MAG-History	7.16e-04	1.69	2.72	3.20	5.11	2.24	5.82	1.50	2.49	6.30	3.40	1.66	1.88	1.35	6.75
coauth-MAG-Geology	3.35e-03	2.01	1.97	1.69	2.43	1.84	2.71	1.31	0.97	1.99	1.74	1.06	1.26	0.94	4.74
music-rap-genius	6.82e-04	5.44	6.92	1.98	1.85	1.62	2.10	1.82	2.15	1.93	2.00	1.78	2.09	1.39	2.67
tags-stack-overflow	1.84e-04	13.08	10.42	3.97	6.45	9.43	6.63	3.37	2.74	2.95	3.60	1.08	1.85	–	3.37
tags-math-sx	1.08e-03	9.08	8.67	2.88	6.19	9.37	6.34	3.48	2.81	4.53	2.71	1.19	1.55	1.86	13.99
tags-ask-ubuntu	1.08e-03	12.29	12.64	4.24	7.15	4.96	7.51	7.48	5.63	7.10	4.15	1.75	2.54	1.19	7.48
threads-stack-overflow	1.14e-05	23.85	31.12	12.97	2.73	3.85	3.19	5.20	3.89	1.06	11.54	1.66	4.06	–	1.53
threads-math-sx	5.63e-05	20.86	16.01	5.03	25.08	28.13	23.32	10.46	7.46	11.04	4.86	0.90	1.18	0.61	47.18
threads-ask-ubuntu	1.31e-04	78.12	80.94	29.00	21.04	2.80	30.82	7.09	6.62	16.63	32.31	0.94	1.51	1.78	9.82
NDC-substances	1.17e-03	4.90	5.27	2.90	5.92	3.36	5.97	4.76	4.46	5.35	2.93	1.39	1.83	1.86	8.17
NDC-classes	6.72e-03	4.43	3.38	1.82	1.27	1.19	0.99	0.94	2.14	0.92	1.34	0.78	0.91	2.45	0.62
DAWN	8.47e-03	4.43	3.86	2.13	4.73	3.76	4.77	3.76	1.45	4.61	2.04	1.57	1.37	1.55	2.86
congress-committees	6.99e-04	3.59	3.28	2.48	4.83	2.49	5.04	1.06	1.31	3.21	2.59	1.50	3.89	2.13	7.67
congress-bills	1.71e-04	0.93	0.90	0.88	0.65	1.23	0.66	0.60	0.55	0.60	0.78	3.16	1.07	6.01	107.19
email-Enron	1.40e-02	1.78	1.62	1.33	0.85	0.83	0.87	1.27	0.83	0.99	1.28	3.69	3.16	2.02	0.72
email-Eu	5.34e-03	1.98	2.15	1.78	1.28	2.69	1.37	0.88	1.55	1.01	1.79	1.59	1.75	1.26	3.47
contact-high-school	2.47e-03	3.86	4.16	2.54	1.92	3.61	2.00	0.96	1.13	1.72	2.53	1.39	2.41	0.78	2.86
contact-primary-school	2.59e-03	5.63	6.40	3.96	2.98	2.95	3.21	0.92	0.94	1.63	4.02	1.41	4.31	0.93	6.91

Our proposed algorithms can achieve much higher performance than randomly guessing open triangles to simplicially close (Table 2). As is the case with standard dyadic link prediction, there is not one score function that performs best over all datasets [34, 36]; however, we note some general trends.

First, the simple harmonic and geometric means of edge weights performs well in many datasets, which further highlights the importance of tie strength in predicting simplicial closure. The effectiveness of these purely local measures contrasts with the traditional dyadic link prediction problem, where accounting for long path lengths using, e.g., personalized PageRank scores are more successful. This suggests that the higher-order link prediction problem has fundamentally different structure than traditional link prediction [34].

Surprisingly, the arithmetic mean performs the worst of the three means in all but one dataset (coauth-MAG-History). We further study the performance of these measures using the generalized mean with parameter p as score functions:

$$s_p(i,j,k) = [(W_{ij}^p + W_{jk}^p + W_{ik}^p)/3]^{1/p}, \tag{3}$$

where W_{ab} is the weight between nodes a and b in the projected graph. The harmonic, arithmetic, and geometric means are the special cases where $p = -1$, $p = 1$, and the limit $p \to 0$.

Generally, prediction performance is (i) unimodal in p, (ii) maximized for $p \in [-1, 0]$, and (iii) better for $p < -1$ than for $p > 1$ (Fig. 6). Two exceptions are the NDC classes dataset, where the maximum is achieved for $p < -1$, and the History co-authorship dataset, where p is maximized near 0.5 and performance is better

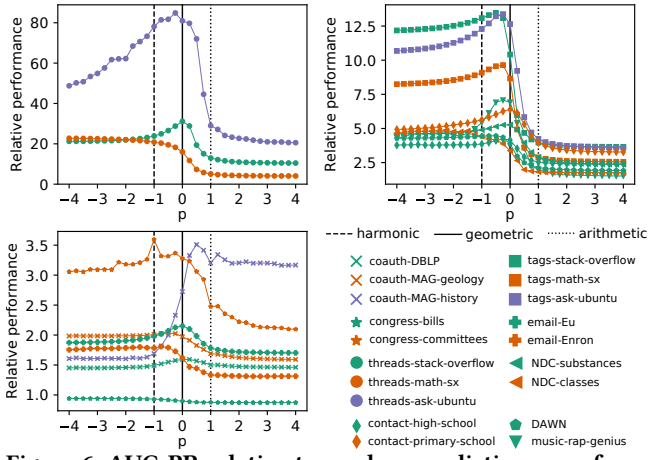


Figure 6: AUC-PR relative to random predictions as a function of the parameter p in the generalized mean score function (Eq. (3)).

for larger values of p than smaller ones. In the former case, the dataset has the unique property that no open triangles with exactly one strong tie close (Fig. 4). Thus, it makes sense that smaller p perform better, as smaller p account relatively more for the minimum edge weight value ($\lim_{p\to-\infty} s_p(i,j,k) = \min(W_{ij}, W_{jk}, W_{ik})$). In the latter case, history co-authorship has the lowest average degree in the projected graph of all datasets by far (Fig. 2D). Therefore, only a single strong edge may be providing the signal for closure,

for which larger p would be a better score function ($\lim_{p\to+\infty} s_p(i,j,k) = \max(W_{ij}, W_{jk}, W_{ik})$).

Next, the supervised learning approach also performs well broadly, especially in the larger datasets such as the co-authorship datasets, which have sufficient training data to learn a good model. This approach on the congress-bills dataset provides extremely good performance. Here, the supervised method captures an interesting feature of this dataset that nodes with small degree and appearing in few simplices are *more* likely to simplicially close. In fact, the negative of the simple preferential attachment score functions based on projected graph degree and simplicial degree perform over 10 and 20 times better than the random baseline.

Finally, there are similar improvements over the baseline for each type of dataset. Several score functions provide substantial improvements over the baseline for datasets built from tags or threads on stack exchange sites but only modest improvements for the email and co-authorship datasets.

5 DISCUSSION

There is a rich history of modeling network data with graphs. This pairwise paradigm has been successful, even though many network datasets carry natural higher-order structure that is not captured by a graph. One reason for the discrepancy is a lack of evaluation frameworks for higher-order network models. Our work fills this gap by considering a higher-order link prediction problem on the evolution of higher-order network structure: determining which groups of nodes will interact simultaneously in the future, given that they have not done so already. In addition, higher-order link prediction is a tool immediately applicable to domain applications.

We find rich variety in our datasets in terms of the fraction of open triangles and structural properties of the projected graph such as edge density and average degree, although datasets from the same domain tend to have similar structure. Prior research has identified the distinction between open and closed triangles when projecting bipartite networks onto one mode [11, 44, 45], but has not studied the prediction of simplicial closure of the open triangles. Patania et al. measured the fraction of triangles that are open in co-authorship networks [50]; our results are consistent but illuminate that open triangles may be extremely common in other domains.

We also find that principles from the temporal network evolution continue to hold when looking at higher-order structure, namely, edge density and tie strength are positive indicators of simplicial closure for both 3- and 4-node simplices. However, there is tension between these features — the more influential one depends on the dataset domain, suggesting different mechanisms for simplicial closure. We use these principles to develop effective methods for higher-order link prediction. Local measures that only account for the tie strength between nodes in an open triangle perform well, which differs from traditional link prediction where long paths are important [34]. This suggests that higher-order temporal evolution is fundamentally different than traditional network evolution.

Higher-order modeling also opens the door to new methods in network science. For example, we employed ideas from computational topology, a field with recent success in data analysis [13] that directly studies higher-order structure. While much prior research in topological data analysis focuses on persistent homology (which is largely orthogonal to the concepts in our work), there are several recent ideas connecting random walks and simplicial complexes [27, 40, 49, 60]. As we have seen, our higher-order link prediction problem provides a framework for evaluating such ideas on a concrete task.

Analyzing larger simplices brings several challenges, specifically in interpretability of results and in computational scalability. The space of open 4-node configurations containing at least one triangle (Fig. 5A) is already much more complex than the space of open 3-node configurations (Fig. 4A). Comprehending and evaluating the large space of 5-node configurations is difficult. The simplicial PageRank method is already computationally taxing for the two largest datasets we studied. We expect that randomized algorithms providing an approximate solution with less computation will be useful for this task. Regardless, the computational challenges bring exciting avenues for future research.

Software for simplicial closure, higher-order link prediction, and reproducing the results in this paper are available at
`https://github.com/arbenson/ScHoLP-Tutorial`.

ACKNOWLEDGEMENTS

We thank Mason Porter and Peter Mucha for providing the congress committees dataset. This research was supported in part by a Simons Investigator Award. RA is also supported in part by a Google scholarship and a Facebook scholarship. AJ received funding from the Vannevar Bush Fellowship from the office of the Secretary of Defense. MTS received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 702410. The funders had no role in the design of this study; the results presented here reflect solely the authors' views.

REFERENCES

[1] Lada A Adamic and Eytan Adar. 2003. Friends and neighbors on the web. *Social networks* 25, 3 (2003), 211–230.

[2] Réka Albert and Albert-László Barabási. 2002. Statistical mechanics of complex networks. *Reviews of Modern Physics* 74, 1 (jan 2002), 47–97. https://doi.org/10.1103/revmodphys.74.47

[3] Lars Backstrom, Dan Huttenlocher, Jon Kleinberg, and Xiangyang Lan. 2006. Group formation in large social networks: membership, growth, and evolution. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 44–54.

[4] Lars Backstrom and Jure Leskovec. 2011. Supervised Random Walks: Predicting and Recommending Links in Social Networks. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (WSDM '11)*. ACM, New York, NY, USA, 635–644. https://doi.org/10.1145/1935826.1935914

[5] Bahman Bahmani, Abdur Chowdhury, and Ashish Goel. 2010. Fast incremental and personalized PageRank. *Proceedings of the VLDB Endowment* 4, 3 (2010), 173–184.

[6] Grey Ballard, Tamara G Kolda, Ali Pinar, and C Seshadhri. 2015. Diamond sampling for approximate maximum all-pairs dot-product (MAD) search. In *IEEE International Conference on Data Mining*. IEEE, 11–20.

[7] A.L Barabási, H Jeong, Z Néda, E Ravasz, A Schubert, and T Vicsek. 2002. Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications* 311, 3-4 (aug 2002), 590–614. https://doi.org/10.1016/s0378-4371(02)00736-7

[8] A. R. Benson, D. F. Gleich, and J. Leskovec. 2016. Higher-order organization of complex networks. *Science* 353, 6295 (jul 2016), 163–166. https://doi.org/10.1126/science.aad9029

[9] Claude Berge. 1989. *Hypergraphs*. Elsevier.

[10] Norman Biggs. 1974. *Algebraic Graph Theory*. Cambridge University Press. https://doi.org/10.1017/cbo9780511608704

[11] Jason Cory Brunson. 2015. Triadic analysis of affiliation networks. *Network Science* 3, 4 (2015), 480–508.

[12] Ed Bullmore and Olaf Sporns. 2009. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience* 10, 3 (2009), 186.

[13] Gunnar Carlsson. 2009. Topology and data. *Bull. Amer. Math. Soc.* 46, 2 (jan 2009), 255–308. https://doi.org/10.1090/s0273-0979-09-01249-x

[14] Norishige Chiba and Takao Nishizeki. 1985. Arboricity and subgraph listing algorithms. *SIAM J. Comput.* 14, 1 (1985), 210–223.

[15] Jesse Davis and Mark Goadrich. 2006. The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning*. 233–240.

[16] Charlotte M. Deane, Łukasz Salwiński, Ioannis Xenarios, and David Eisenberg. 2002. Protein Interactions: two methods for assessment of the reliability of high throughput observations. *Molecular & Cellular Proteomics* 1, 5 (apr 2002), 349–356. https://doi.org/10.1074/mcp.m100037-mcp200

[17] David Easley and Jon Kleinberg. 2010. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press.

[18] Scott L. Feld. 1981. The focused organization of social ties. *Amer. J. Sociology* 86, 5 (1981).

[19] James H. Fowler. 2006. Connecting the Congress: A Study of Cosponsorship Networks. *Political Analysis* 14, 04 (2006), 456–487. https://doi.org/10.1093/pan/mpl002

[20] James H. Fowler. 2006. Legislative cosponsorship networks in the US House and Senate. *Social Networks* 28, 4 (oct 2006), 454–465. https://doi.org/10.1016/j.socnet.2005.11.003

[21] Peter Frankl. 1995. Extremal set systems. In *Handbook of combinatorics*, Ron Graham, Martin Groetschel, and Laszlo Lovasz (Eds.). Vol. 1. Elsevier.

[22] Mark S Granovetter. 1973. The strength of weak ties. *Amer. J. Sociology* 78, 6 (1973), 1360–1380.

[23] Jacopo Grilli, György Barabás, Matthew J. Michalska-Smith, and Stefano Allesina. 2017. Higher-order interactions stabilize dynamics in competitive network models. *Nature* (jul 2017). https://doi.org/10.1038/nature23273

[24] Allen Hatcher. 2002. *Algebraic topology*. Cambridge University Press.

[25] Willem J Heiser and Mohammed Bennani. 1997. Triadic distance models: axiomatization and least squares representation. *Journal of Mathematical Psychology* 41, 2 (1997), 189–206.

[26] Petter Holme and Jari Saramäki. 2012. Temporal networks. *Physics reports* 519, 3 (2012), 97–125.

[27] Paul Horn, Ali Jadbabaie, and Gabor Lippner. 2017. Edge ranking in simplicial complexes via page-rank. (2017). In preparation.

[28] Leo Katz. 1953. A new status index derived from sociometric analysis. *Psychometrika* 18, 1 (1953), 39–43.

[29] Bryan Klimt and Yiming Yang. 2004. Introducing the Enron Corpus. In *CEAS*.

[30] Gueorgi Kossinets and Duncan J Watts. 2006. Empirical analysis of an evolving social network. *Science* 311, 5757 (2006), 88–90.

[31] Matthieu Latapy. 2008. Main-memory triangle computations for very large (sparse (power-law)) graphs. *Theoretical Computer Science* 407, 1-3 (2008), 458–473.

[32] Jure Leskovec, Lars Backstrom, Ravi Kumar, and Andrew Tomkins. 2008. Microscopic evolution of social networks. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM Press. https://doi.org/10.1145/1401890.1401948

[33] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Signed networks in social media. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 1361–1370.

[34] David Liben-Nowell and Jon Kleinberg. 2007. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology* 58, 7 (2007), 1019–1031. https://doi.org/10.1002/asi.20591

[35] Lek-Heng Lim. 2015. Hodge Laplacians on graphs. In *Proceedings of Symposia in Applied Mathematics, Geometry and Topology in Statistical Inference*, Sayan Mukherjee (Ed.), Vol. 73. AMS.

[36] Linyuan Lü and Tao Zhou. 2011. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications* 390, 6 (mar 2011), 1150–1170. https://doi.org/10.1016/j.physa.2010.11.027

[37] Rossana Mastrandrea, Julie Fournet, and Alain Barrat. 2015. Contact patterns in a high school: a comparison between data collected using wearable sensors, contact diaries and friendship surveys. *PloS one* 10, 9 (2015), e0136497.

[38] R. Milo. 2002. Network Motifs: Simple Building Blocks of Complex Networks. *Science* 298, 5594 (oct 2002), 824–827. https://doi.org/10.1126/science.298.5594.824

[39] Yves Moreau and Léon-Charles Tranchevent. 2012. Computational tools for prioritizing candidate genes: boosting disease gene discovery. *Nature Reviews Genetics* 13, 8 (jul 2012), 523–536. https://doi.org/10.1038/nrg3253

[40] Sayan Mukherjee and John Steenbergen. 2016. Random walks on simplicial complexes and harmonics. *Random Structures & Algorithms* 49, 2 (mar 2016), 379–405. https://doi.org/10.1002/rsa.20645

[41] Saket Navlakha and Carl Kingsford. 2010. The power of protein interaction networks for associating genes with diseases. *Bioinformatics* 26, 8 (2010), 1057–1063.

[42] Mark EJ Newman. 2001. Clustering and preferential attachment in growing networks. *Physical review E* 64, 2 (2001), 025102.

[43] M. E. J. Newman. 2003. The Structure and Function of Complex Networks. *SIAM Rev.* 45, 2 (jan 2003), 167–256. https://doi.org/10.1137/s003614450342480

[44] M. E. J. Newman, Duncan J. Watts, and Steven H. Strogatz. 2002. Random graph models of social networks. *Proceedings of the National Academy of Sciences* 99, Suppl.1 (2002).

[45] Tore Opsahl. 2013. Triadic closure in two-mode networks: Redefining the global and local clustering coefficients. *Social Networks* 35, 2 (may 2013), 159–167. https://doi.org/10.1016/j.socnet.2011.07.001

[46] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank Citation Ranking: Bringing Order to the Web*. Technical Report 1999-66. Stanford University. http://dbpubs.stanford.edu:8090/pub/1999-66

[47] Christopher C. Paige and Michael A. Saunders. 1982. LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares. *ACM Trans. Math. Software* 8, 1 (mar 1982), 43–71. https://doi.org/10.1145/355984.355989

[48] Ashwin Paranjape, Austin R Benson, and Jure Leskovec. 2017. Motifs in temporal networks. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM, 601–610.

[49] Ori Parzanchevski and Ron Rosenthal. 2016. Simplicial complexes: Spectrum, homology and random walks. *Random Structures & Algorithms* 50, 2 (may 2016), 225–261. https://doi.org/10.1002/rsa.20657

[50] Alice Patania, Giovanni Petri, and Francesco Vaccarino. 2017. The shape of collaborations. *EPJ Data Science* 6, 1 (aug 2017). https://doi.org/10.1140/epjds/s13688-017-0114-8

[51] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[52] Ali Pinar, C. Seshadhri, and Vaidyanathan Vishal. 2017. ESCAPE: Efficiently Counting All 5-Vertex Subgraphs. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 1431–1440. https://doi.org/10.1145/3038912.3052597

[53] Mason A. Porter, Peter J. Mucha, M.E.J. Newman, and A.J. Friend. 2007. Community structure in the United States House of Representatives. *Physica A: Statistical Mechanics and its Applications* 386, 1 (dec 2007), 414–438. https://doi.org/10.1016/j.physa.2007.07.039

[54] M. A. Porter, P. J. Mucha, M. E. J. Newman, and C. M. Warmbrand. 2005. A network analysis of committees in the U.S. House of Representatives. *Proceedings of the National Academy of Sciences* 102, 20 (may 2005), 7057–7062. https://doi.org/10.1073/pnas.0500191102

[55] Anatol Rapoport. 1953. Spread of information through a population with socio-structural bias I: Assumption of transitivity. *Bulletin of Mathematical Biology* 15, 4 (1953), 523–533.

[56] Mikail Rubinov and Olaf Sporns. 2010. Complex network measures of brain connectivity: uses and interpretations. *Neuroimage* 52, 3 (2010), 1059–1069.

[57] Yousef Saad and Martin H. Schultz. 1986. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM J. Sci. Statist. Comput.* 7, 3 (jul 1986), 856–869. https://doi.org/10.1137/0907058

[58] Aneesh Sharma, C Seshadhri, and Ashish Goel. 2017. When Hashes Met Wedges: A Distributed Algorithm for Finding High Similarity Vectors. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 431–440.

[59] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-june Paul Hsu, and Kuansan Wang. 2015. An overview of Microsoft Academic Service (MAS) and applications. In *Proceedings of the 24th international conference on world wide web*. ACM, 243–246.

[60] John Steenbergen, Caroline Klivans, and Sayan Mukherjee. 2014. A Cheeger-type inequality on simplicial complexes. *Advances in Applied Mathematics* 56 (may 2014), 56–77. https://doi.org/10.1016/j.aam.2014.01.002

[61] Juliette Stehlé, Nicolas Voirin, Alain Barrat, Ciro Cattuto, Lorenzo Isella, Jean-François Pinton, Marco Quaggiotto, Wouter Van den Broeck, Corinne Régis, Bruno Lina, et al. 2011. High-resolution measurements of face-to-face contact patterns in a primary school. *PloS one* 6, 8 (2011), e23176.

[62] Jie Tang, Sen Wu, Jimeng Sun, and Hang Su. 2012. Cross-domain collaboration recommendation. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM Press. https://doi.org/10.1145/2339530.2339730

[63] J. Ugander, L. Backstrom, C. Marlow, and J. Kleinberg. 2012. Structural diversity in social contagion. *Proceedings of the National Academy of Sciences* 109, 16 (apr 2012), 5962–5966. https://doi.org/10.1073/pnas.1116502109

[64] Chao Wang, Venu Satuluri, and Srinivasan Parthasarathy. 2007. Local probabilistic models for link prediction. In *Seventh IEEE International Conference on Data*

Mining. IEEE, 322–331.

[65] Dashun Wang, Dino Pedreschi, Chaoming Song, Fosca Giannotti, and Albert-Laszlo Barabasi. 2011. Human mobility, social ties, and link prediction. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM Press. https://doi.org/10.1145/2020408.2020581

[66] X. Wang, N. Gulbahce, and H. Yu. 2011. Network-based methods for human disease gene prediction. *Briefings in Functional Genomics* 10, 5 (jul 2011), 280–293. https://doi.org/10.1093/bfgp/elr024

A DATASET DESCRIPTIONS

Here we provide a more complete description of the datasets used in the main text. All datasets except the U.S. congress committee membership and music collaboration datasets are available at

http://www.cs.cornell.edu/~arb/data/.

Co-authorship. In these datasets, the nodes correspond to authors, and each simplex represents the authors on a scientific publication. The timestamp is the year of publication. We analyze three co-authorship networks—one derived from DBLP, an online bibliography for computer science, and two derived from the Microsoft Academic Graph. We used the September 3, 2017 release of DBLP[2] and the MAG version released with the Open Academic Graph[3] [59]. We constructed two field-specific datasets by filtering the data according to keywords in the "field of study" information. One dataset consisted of all papers with "History" as a field of study and the other all papers with "Geology" as a field of study.

Stack exchange tags. Stack exchange is a collection of question-and-answer web sites.[4] Users post questions and may annotate each question with up to 5 tags that specify topic areas spanned by the question. We derive tag networks where nodes correspond to tags and each simplex represents the tags on a question. The timestamp for a simplex is the time that the question was posted on the web site. We derived three datasets corresponding to three stack exchange web sites:

(1) https://stackoverflow.com,
(2) https://math.stackexchange.com, and
(3) https://askubuntu.com

. The raw data was downloaded from the Stack Exchange data dump[5] (downloaded September 20, 2017), which contains the entire history of the content on the stack exchange web sites.

Stack exchange threads. We also formed user interaction datasets from the stack exchange web sites. Users post answers to questions, creating a question-and-answer "thread." We constructed datasets where the nodes are users and simplices correspond to the users asking a question or posting an answer on a single thread. We only considered threads where the question and all answers were posted within 24 hours. The timestamps of the simplices are the times that the question was posted.

National Drug Code Directory (NDC). Under the Drug Listing Act of 1972, the U.S. Food and Drug Administration releases information on all commercial drugs going through the regulation of the agency. We constructed two datasets from this data where simplices correspond to drugs. In one, the nodes are classification labels (e.g., serotonin reuptake inhibitor), and simplices are comprised of all labels applied to a drug; in the other, the nodes are substances (e.g., testosterone) and simplices are constructed from all substances in a drug. In both derived datasets, the timestamps are the days when the drugs were first marketed.

United States Congress. We derived two datasets from political networks, where the nodes are congresspersons in the U.S. congress. In the first, simplices represent all members of committees and sub-committees in the House of Representatives (Congresses 101 to 107, from 1989 to 2003), and the timestamp of the simplex is the year that the committee formed [53, 54]. In the second dataset, simplices are comprised of the sponsor and co-sponsors of legislative bills put forth in both the House of Representatives and the Senate [19, 20], and the timestamps are the days that the bills were introduced.

Email. In email communication, messages can be sent to multiple recipients. We analyze two email datasets—one from communication between Enron employees [29] and the other from a European research institution [48]. In both datasets, nodes are email addresses. In the Enron dataset, a simplex consists of the sender and all recipients of the email. The data source for the European research institution only contains (sender, receiver, timestamp) tuples, where timestamps are recorded at 1-second resolution [48]. Simplices consist of a sender and all receivers such that the email between the two has the same timestamp.

Human contact. The human contact networks are constructed from interactions recorded by wearable sensors in a high school [37] and a primary school [61]. The sensors record proximity-based contacts every 20 seconds. We construct a graph for each interval, where nodes i and j are connected if they are in contact during the interval. We then consider simplices to be all maximal cliques in the graph at each time interval.

DAWN. The Drug Abuse Warning Network (DAWN) is a national health surveillance system that records drug use contributing to hospital emergency department visits throughout the United States. Simplices in our dataset are the drugs used by a patient (as reported by the patient) in an emergency department visit. The drugs include illicit substances, prescription and over-the-counter medication, and dietary supplements. Timestamps of visits are recorded at the resolution of quarter-years, spanning a total duration of 8 years. For a period of time, the recording system only recorded the first 16 drugs reported by a patient, so we only use (at most) the first 16 drugs reported by a patient for the entire dataset.

Music collaboration. Musical artists often collaborate on individual songs. We derive a dataset where nodes are artists and a simplex consists of all artists collaborating on a song. The songs were obtained from a web crawl of the genius.com music lyrics web site[6]. We consider the collaborating artists to be the lead artist along with any "featured" artists (this excludes some cases where lyrics from an artist are included but that artist is not listed as a featured artist). The timestamps are the release dates of the song. We only collected data from songs that contain the "rap" tag on the web site and discarded songs without a specified release date. The crawler ran for several days and collected over 500,000 songs.

[2] http://dblp.org/xml/release/
[3] https://www.openacademic.ai/oag/
[4] https://stackexchange.com
[5] https://archive.org/details/stackexchange
[6] https://genius.com/

B TEMPORAL ASYNCHRONICITY AND OPEN TRIANGLES

Our datasets contain temporal dynamics, so edges may only be "active" for certain periods in the overall timespan of the dataset. This provides one plausible explanation for the formation of open triangles. For example, in co-authorship networks, an open triangle may arise when three separate collaborations occurred in disjoint time periods. To investigate the importance of such effects, we analyze the asynchronicity in open triangles in our datasets. We define the "active interval" of an edge in the projected graph to be the interval bounded by the earliest and latest timestamps of simplices containing the two nodes in the edge. Recall that our datasets are defined by a collection of timestamped simplices $\{(S_i, t_i)\}$, where each $S_i \subset V$ is the simplex and each $t_i \in \mathbb{R}$ is a timestamp. The active interval of an edge (u, v) is then

$$I_{u,v} = [\min\{t_i \mid u, v \in S_i\},\ \max\{t_i \mid u, v \in S_i\}]. \tag{4}$$

For each open triangle in each dataset, we compute the number of pairwise overlapping active intervals amongst the three edges in the triangle (Table 3). In the majority of cases, all three pairs of intervals overlap. By Helly's theorem, this implies that most of the time, there is an interval of time for which all three edges are simultaneously active. Stated differently, in the co-authorship example, the collaborators could have theoretically formed a closed triangle during this time period, but they did not. We conclude that temporal asynchronicity is not a major reason for the presence of open triangles in our datasets.

Table 3: Temporal asynchronicity and open triangles. For each open triangle in each dataset, we find the number of overlaps between the active intervals of the three edges, where an active interval of an edge has end points given by the earliest and latest timestamps of simplices containing the two nodes in the edges (Eq. (4)). The edges in most open triangles have three pairwise overlapping intervals. In these cases, there is a time period where all three edges were simultaneously active by Helly's theorem.

		# overlaps			
Dataset	# open triangles	0	1	2	3
coauth-DBLP	1,295,214	0.012	0.143	0.123	0.722
coauth-MAG-history	96,420	0.002	0.055	0.059	0.884
coauth-MAG-geology	2,494,960	0.010	0.128	0.109	0.753
tags-stack-overflow	300,646,440	0.002	0.067	0.071	0.860
tags-math-sx	2,666,353	0.001	0.040	0.049	0.910
tags-ask-ubuntu	3,288,058	0.002	0.088	0.085	0.825
threads-stack-overflow	99,027,304	0.001	0.034	0.037	0.929
threads-math-sx	11,294,665	0.001	0.038	0.039	0.922
threads-ask-ubuntu	136,374	0.000	0.020	0.023	0.957
NDC-substances	1,136,357	0.020	0.196	0.151	0.633
NDC-classes	9,064	0.022	0.191	0.136	0.652
DAWN	5,682,552	0.027	0.216	0.155	0.602
congress-committees	190,054	0.001	0.046	0.058	0.895
congress-bills	44,857,465	0.003	0.063	0.113	0.821
email-Enron	3,317	0.008	0.130	0.151	0.711
email-Eu	234,600	0.010	0.131	0.132	0.727
contact-high-school	31,850	0.000	0.015	0.019	0.966
contact-primary-school	98,621	0.000	0.012	0.014	0.974
music-rap-genius	70,057	0.028	0.221	0.141	0.611

C SIMPLICIAL CLOSURE AT DIFFERENT POINTS IN TIME

In the main text, we studied simplicial closure by counting the 3-node and 4-node configuration patterns in the first 80% of timestamped simplices and then measuring the fraction of instances that simplicially close in the final 20% of timestamped simplices. Here, we show that our results are robust when examining different time slices of the data. We first filtered each dataset to contain only the first X% of timestamped simplices, for $X = 40, 60, 80$. (The original dataset is the case of $X = 100$.) We then split the filtered dataset into the first 80% and last 20% of timestamped simplices (within the time frame of the filtered dataset) and computed simplicial closure probabilities.

Table 5 lists the 3-node simplicial closure probabilities as a function of the configuration of the 3 nodes in the first 80% of the data for $X = 40, 60, 80, 100$. Broadly, the closure probabilities remain similar for different values of X. We also find that edge density and tie strength are always positive indicators of simplicial closure (Table 4), regardless of X. Thus, these features are important for simplicial closure throughout the time spanned by the datasets.

The tension between these features is also consistent over time. The weak open triangle (where all three edges are weak ties) is more likely to close than the strong wedge (the 3-node configuration with exactly two strong ties) in the coauth-DBLP, coauth-MAG-Geology, and congress-bills datasets for all values of X as well as in the congress-committees dataset for $X = 60, 80, 100$. On the other hand, the strong wedge is more likely to close in the three stack exchange tags networks, DAWN, and threads-stack-overflow for all values of X as well as in threads-math-sx for $X = 80, 100$.

Table 4: Robustness in dependence of tie strength and edge density in 3-node configurations at different time slices of the data. For edge density, we tested whether or not the closure probability of a fixed weighted induced subgraph configuration and the same configuration with an additional unit-weight edge significantly increases or decreases the closure probability (at significance level 10^{-5}). For tie strength, we tested whether the closure probability significantly increases or decreases when comparing a fixed weighted induced subgraph containing at least one weak tie, and the same configuration where the weak tie is converted to a strong tie (edge weight at least 2 in the projected graph). The "total" column is the number of tested hypotheses. We apply the tests to filtered datasets that only contain the first X% of timestamped simplices (in time order). We also only consider cases where the configuration has at least 25 samples in the first 80% of timestamped simplices of a filtered dataset. Increasing either edge density or tie strength significantly increases the closure probability for all values of X, suggesting that these features are positive indicators of simplicial closure over time.

	edge density increases			tie strength increases		
X%	sig. incr.	sig. decr.	total	sig. incr.	sig. decr.	total
40	89	0	113	71	2	113
60	101	0	113	80	7	113
80	102	0	113	86	2	113
100	96	0	107	76	6	107

Table 5: Closure probabilities of different configurations at different points in time. We first filtered each dataset to contain only the first X% of timestamped simplices, for $X = 40, 60, 80, 100$. We then split the filtered dataset into the first 80% and last 20% of timestamped simplices (within the time frame of the filtered dataset). We record the probability of closure in last 20% conditioned on the open configuration in the first 80%.

																								
	40	60	80	100	40	60	80	100	40	60	80	100	40	60	80	100	40	60	80	100	40	60	80	100
coauth-DBLP	8.2e-13	1.2e-12	8.3e-13	9.3e-13	3.1e-08	4.2e-08	3.4e-08	3.6e-08	1.1e-07	1.5e-07	1.2e-07	1.3e-07	4.4e-04	5.2e-04	3.8e-04	3.5e-04	9.7e-04	1.2e-03	9.4e-04	8.8e-04	2.0e-03	2.5e-03	2.2e-03	2.1e-03
coauth-MAG-Geology	7.9e-12	5.0e-12	3.2e-12	4.2e-12	1.1e-07	9.9e-08	7.6e-08	8.9e-08	4.1e-07	4.6e-07	3.6e-07	4.5e-07	5.8e-04	6.8e-04	5.1e-04	5.3e-04	1.4e-03	1.8e-03	1.5e-03	1.6e-03	3.0e-03	4.4e-03	3.7e-03	4.1e-03
coauth-MAG-History	1.2e-12	8.8e-13	3.3e-13	1.8e-13	2.7e-08	2.0e-08	1.0e-08	5.6e-09	1.6e-07	1.4e-07	6.3e-08	3.9e-08	1.4e-04	1.8e-04	1.0e-04	6.3e-05	6.2e-04	8.5e-04	4.1e-04	2.9e-04	1.3e-03	2.3e-03	2.5e-03	1.0e-03
music-rap-genius	1.6e-09	8.6e-10	4.3e-10	1.2e-10	1.3e-06	6.2e-07	5.7e-07	2.3e-07	5.5e-06	2.9e-06	1.9e-06	1.1e-06	3.3e-04	2.2e-04	2.2e-04	1.3e-04	1.0e-03	8.0e-04	5.6e-04	4.4e-04	3.3e-03	2.9e-03	1.7e-03	1.3e-03
tags-stack-overflow	3.6e-09	3.5e-09	2.9e-09	2.8e-09	4.8e-07	4.5e-07	3.8e-07	3.8e-07	5.0e-06	4.6e-06	4.2e-06	4.2e-06	9.6e-06	8.5e-06	7.4e-06	7.4e-06	6.9e-05	6.3e-05	5.8e-05	5.7e-05	4.6e-04	4.1e-04	3.8e-04	3.7e-04
tags-math-sx	1.0e-06	1.4e-06	1.9e-06	1.9e-06	1.7e-05	1.7e-05	1.9e-05	2.1e-05	1.1e-04	1.1e-04	1.5e-04	1.4e-04	1.3e-04	1.2e-04	1.3e-04	1.2e-04	7.0e-04	7.0e-04	7.2e-04	6.9e-04	3.4e-03	3.3e-03	3.4e-03	3.2e-03
tags-ask-ubuntu	4.9e-07	5.4e-07	7.9e-07	4.8e-07	1.1e-05	1.3e-05	1.4e-05	9.8e-06	8.1e-05	9.7e-05	1.1e-04	7.9e-05	9.4e-05	8.8e-05	8.6e-05	6.2e-05	4.8e-04	4.6e-04	4.7e-04	3.8e-04	1.5e-03	1.6e-03	1.5e-03	1.4e-03
threads-stack-overflow	3.2e-12	8.4e-13	4.3e-13	2.2e-13	5.5e-09	2.2e-09	1.4e-09	9.0e-10	8.4e-08	4.2e-08	2.7e-08	2.1e-08	6.4e-07	3.3e-07	2.5e-07	1.9e-07	4.8e-06	3.0e-06	2.3e-06	2.0e-06	2.5e-05	1.7e-05	1.5e-05	1.5e-05
threads-math-sx	2.3e-10	1.4e-10	6.4e-11	4.2e-11	1.5e-07	1.3e-07	7.8e-08	6.0e-08	1.2e-06	1.3e-06	9.4e-07	7.4e-07	3.6e-06	3.8e-06	2.5e-06	2.3e-06	1.6e-05	2.0e-05	1.7e-05	1.5e-05	6.4e-05	8.9e-05	9.0e-05	7.9e-05
threads-ask-ubuntu	5.3e-11	1.4e-11	5.3e-12	3.2e-12	2.8e-08	2.1e-08	1.2e-08	9.0e-09	5.7e-07	5.2e-07	5.7e-07	4.1e-07	1.7e-06	5.6e-07	7.7e-07	7.8e-07	1.6e-05	1.4e-05	1.8e-05	1.6e-05	6.8e-05	1.0e-04	1.6e-04	1.4e-04
NDC-substances	1.4e-06	3.4e-06	9.6e-07	3.8e-07	1.1e-04	1.9e-04	7.5e-05	3.3e-05	1.9e-04	4.2e-04	1.9e-04	7.8e-05	2.5e-03	2.9e-03	1.4e-03	4.8e-04	6.0e-03	6.0e-03	3.2e-03	1.1e-03	1.0e-02	1.2e-02	6.7e-03	2.2e-03
NDC-classes	5.3e-07	9.0e-06	1.5e-06	8.4e-07	3.7e-06	8.8e-04	1.7e-04	3.1e-04	3.6e-04	1.8e-03	7.7e-04	3.4e-04	0.0e+00	4.0e-02	0.0e+00	4.8e-03	0.0e+00	7.1e-02	3.0e-03	4.8e-03	9.1e-03	5.4e-02	2.4e-02	1.1e-02
DAWN	1.5e-06	2.1e-06	2.7e-06	2.5e-06	4.2e-05	4.6e-05	6.0e-05	5.4e-05	2.1e-04	2.7e-04	3.5e-04	3.4e-04	3.4e-04	3.9e-04	4.7e-04	4.1e-04	1.6e-03	1.8e-03	2.2e-03	2.1e-03	5.2e-03	6.3e-03	7.6e-03	7.3e-03
congress-bills	1.9e-04	9.2e-04	3.0e-04	2.5e-04	7.6e-04	3.0e-03	1.2e-03	1.3e-03	9.9e-04	2.4e-03	1.2e-03	9.1e-04	4.2e-03	1.2e-02	5.4e-03	8.1e-03	5.4e-03	1.0e-02	5.2e-03	6.1e-03	5.0e-03	7.2e-03	3.4e-03	3.8e-03
congress-committees	0.0e+00	1.5e-04	3.7e-05	5.8e-05	0.0e+00	6.7e-04	2.9e-04	3.6e-04	0.0e+00	9.9e-04	5.7e-04	6.3e-04	0.0e+00	2.4e-03	1.5e-03	1.7e-03	0.0e+00	3.2e-03	2.2e-03	2.1e-03	0.0e+00	3.4e-03	3.0e-03	3.1e-03
email-Eu	8.2e-06	1.5e-05	1.4e-05	8.4e-06	1.3e-04	1.8e-04	1.4e-04	8.3e-05	3.3e-04	3.6e-04	5.0e-04	2.4e-04	1.7e-03	1.1e-03	1.2e-03	1.0e-03	3.6e-03	3.3e-03	3.9e-03	2.4e-03	7.8e-03	6.4e-03	8.1e-03	5.2e-03
email-Enron	6.3e-04	4.3e-04	3.8e-04	3.4e-04	5.4e-03	2.2e-03	1.8e-03	1.9e-03	4.1e-03	4.3e-03	3.3e-03	3.1e-03	1.9e-02	1.1e-02	1.5e-02	9.4e-03	2.4e-02	2.6e-02	2.3e-02	1.2e-02	2.4e-02	3.9e-02	2.5e-02	2.1e-02
contact-high-school	6.4e-07	1.1e-06	1.1e-06	9.4e-07	1.5e-05	1.6e-05	7.5e-06	1.2e-05	5.3e-05	4.3e-05	8.6e-05	3.7e-05	3.8e-04	0.0e+00	9.1e-05	7.2e-05	1.1e-03	4.1e-04	6.7e-04	3.5e-04	2.4e-03	1.6e-03	2.1e-03	1.4e-03
contact-primary-school	2.6e-06	0.0e+00	3.2e-05	1.0e-06	3.7e-06	1.7e-05	6.9e-05	3.2e-06	6.7e-05	5.6e-05	3.2e-04	1.0e-04	9.0e-05	0.0e+00	1.9e-04	5.1e-05	2.7e-04	2.6e-04	8.6e-04	2.6e-04	1.4e-03	9.9e-04	2.1e-03	8.8e-04

																
	40	60	80	100	40	60	80	100	40	60	80	100	40	60	80	100
coauth-DBLP	8.3e-03	8.6e-03	8.5e-03	7.6e-03	1.0e-02	1.1e-02	1.1e-02	1.1e-02	1.2e-02	1.5e-02	1.5e-02	1.7e-02	1.3e-02	1.6e-02	1.7e-02	1.9e-02
coauth-MAG-Geology	8.6e-03	1.2e-02	9.4e-03	1.0e-02	1.4e-02	2.0e-02	1.5e-02	1.7e-02	1.7e-02	2.9e-02	2.3e-02	2.7e-02	2.2e-02	3.7e-02	3.0e-02	4.0e-02
coauth-MAG-History	1.7e-03	3.4e-03	1.6e-03	1.9e-03	6.0e-03	9.1e-03	4.9e-03	3.8e-03	8.3e-03	2.1e-02	1.4e-02	9.1e-03	5.1e-03	3.6e-02	3.8e-02	1.5e-02
music-rap-genius	1.4e-03	2.3e-03	1.3e-03	1.1e-03	4.8e-03	2.9e-03	2.8e-03	2.2e-03	1.2e-02	7.9e-03	6.5e-03	4.6e-03	2.0e-02	1.7e-02	1.2e-02	8.6e-03
tags-stack-overflow	9.2e-05	7.0e-05	6.5e-05	6.5e-05	5.2e-04	4.4e-04	4.0e-04	3.9e-04	2.7e-03	2.3e-03	2.1e-03	2.1e-03	9.6e-03	8.4e-03	7.7e-03	7.6e-03
tags-math-sx	6.3e-04	5.6e-04	5.4e-04	5.6e-04	2.4e-03	2.3e-03	2.3e-03	2.1e-03	1.0e-02	9.1e-03	9.2e-03	8.6e-03	2.9e-02	2.7e-02	2.7e-02	2.6e-02
tags-ask-ubuntu	5.3e-04	4.5e-04	3.3e-04	2.9e-04	2.2e-03	2.2e-03	1.8e-03	1.7e-03	6.9e-03	7.1e-03	5.9e-03	5.9e-03	2.1e-02	2.3e-02	1.8e-02	1.9e-02
threads-stack-overflow	9.6e-06	6.1e-06	5.7e-06	4.9e-06	4.2e-05	3.5e-05	2.6e-05	2.5e-05	1.5e-04	1.3e-04	1.1e-04	1.1e-04	6.9e-04	5.8e-04	4.3e-04	4.7e-04
threads-math-sx	6.3e-05	5.9e-05	4.3e-05	4.0e-05	1.8e-04	2.3e-04	2.0e-04	1.7e-04	4.6e-04	6.4e-04	6.1e-04	5.4e-04	1.3e-03	2.3e-03	2.7e-03	2.2e-03
threads-ask-ubuntu	0.0e+00	0.0e+00	0.0e+00	8.4e-05	4.5e-05	9.2e-05	3.8e-04	2.9e-04	0.0e+00	5.2e-04	1.7e-03	6.5e-04	1.4e-03	2.2e-03	5.7e-03	3.6e-03
NDC-substances	1.5e-02	1.1e-02	7.6e-03	2.1e-03	3.1e-02	2.2e-02	1.3e-02	3.8e-03	4.8e-02	3.5e-02	2.2e-02	7.2e-03	7.3e-02	4.8e-02	3.9e-02	1.5e-02
NDC-classes	0.0e+00	0.0e+00	0.0e+00	0.0e+00	0.0e+00	0.0e+00	1.1e-01	0.0e+00	0.0e+00	1.3e-01	9.1e-02	5.2e-02	3.6e-02	0.0e+00	8.7e-02	3.4e-02
DAWN	1.7e-03	1.5e-03	2.3e-03	1.7e-03	5.7e-03	6.0e-03	7.9e-03	7.2e-03	1.5e-02	1.7e-02	2.2e-02	2.1e-02	4.8e-02	5.5e-02	7.3e-02	6.9e-02
congress-bills	7.9e-03	1.9e-02	9.5e-03	1.7e-02	9.5e-03	1.8e-02	1.1e-02	1.4e-02	8.4e-03	1.6e-02	1.0e-02	1.1e-02	9.6e-03	1.3e-02	1.1e-02	9.3e-03
congress-committees	0.0e+00	9.8e-03	6.1e-03	5.5e-03	0.0e+00	1.2e-02	8.5e-03	5.9e-03	0.0e+00	1.4e-02	1.1e-02	8.4e-03	0.0e+00	1.2e-02	1.6e-02	1.2e-02
email-Eu	2.5e-03	5.2e-03	1.2e-02	5.3e-03	1.6e-02	9.9e-03	1.5e-02	1.2e-02	2.6e-02	1.8e-02	2.4e-02	2.1e-02	4.7e-02	3.4e-02	4.8e-02	3.3e-02
email-Enron	0.0e+00	2.3e-02	0.0e+00	0.0e+00	6.6e-02	3.9e-02	7.6e-02	3.1e-02	5.9e-02	9.9e-02	8.2e-02	4.8e-02	9.2e-02	8.0e-02	1.4e-01	5.5e-02
contact-high-school	0.0e+00	0.0e+00	0.0e+00	2.6e-03	5.2e-03	2.2e-03	4.0e-03	1.8e-03	7.9e-03	4.9e-03	7.9e-03	5.7e-03	1.3e-02	1.2e-02	1.5e-02	1.2e-02
contact-primary-school	0.0e+00	0.0e+00	5.6e-04	3.8e-04	1.4e-03	1.3e-03	1.3e-03	7.9e-04	3.4e-03	3.6e-03	3.2e-03	3.4e-03	1.7e-02	1.8e-02	1.7e-02	1.7e-02

D EFFICIENT COUNTING OF SIMPLICIAL CLOSURE PROBABILITIES

Recall that for our study, an open configuration on three or four nodes is a set of nodes that have not jointly appeared in a simplex in the training set comprising the first 80% of the timestamped simplices. This subgraph configuration "closes" if the nodes subsequently all appear in one of the final 20% of timestamped simplices (the test set). For all newly formed simplices in the test set, we can check their prior configuration c in the training set. This gives us the number of times each configuration closes. Dividing the number of closures of a configuration c by the total number of times it was open in the training set gives the simplicial closure probability. As most of the datasets we consider here are relatively large, however, naively computing the simplicial closure probabilities in Figs. 4 and 5 is infeasible. Hence, we need to develop efficient algorithms for computing the simplicial closure probabilities.

The key idea of our approach is that we do not need to *enumerate* all of the configurations in the training set and check if they close. Instead, we only need the *total count* of open configurations in the training data. We then count how many close by examining the test data directly. The idea of avoiding enumeration when simply counting suffices has been used in other fast graph configuration counting algorithms [48, 52].

D.1 Counting for 3-node configurations

We first show how to count the number of each 3-node subgraph configuration (the top row of Table 6). Recall that a weak tie corresponds to an edge in the projected graph with a weight of 1, whereas a strong tie corresponds to an edge with a weight of at least 2. Subscripts of 1 and 2 denote weak and strong ties in our notation. (Note that we use "2+" for strong ties in the illustrations in Table 6; however, it will be convenient to use the integer 2 in our description of the algorithms.)

Let $\tau_{i,j,k}$, $1 \le i \le j \le k \le 2$, be the number of (open or closed) triangles whose edges have the tie strengths given by the subscripts. For instance, $\tau_{1,1,1}$ is the number of triangles whose edges are all weak ties. Similarly, let $\sigma_{i,j,k}$ be the number of triangles with given tie strengths that are open (see the right-most configurations in the top row block of Table 6). We can count the number of all triangles $\tau_{i,j,k}$ using a number of efficient triangle enumeration algorithms for sparse graphs [31]. For each of these triangles we then determine whether it is closed by examining the entries of a simplex-to-node adjacency matrix (this can be efficiently read out from our set-based data). The difference between the total number of triangles and the number of closed triangles gives us the open triangle counts $\sigma_{i,j,k}$.

Next, consider the number of 2-edge, 3-node induced "wedge" subgraphs. Let the symbols $\omega_{1,1}$, $\omega_{1,2}$, and $\omega_{2,2}$ denote these configurations, where the tie strengths of the two edges are given by the subscripts (see the top row block of Table 6). Furthermore, let $d_1(u)$ and $d_2(u)$ be the number of weak and strong ties containing node u as an endpoint. Then $\omega_{i,j}$ is given by the number of (non-induced) 2-edge, 3-node subgraphs with tie strengths i and j minus the ones that appear in triangles:

$$\omega_{1,1} = \textstyle\sum_u \binom{d_1(u)}{2} - 3\tau_{1,1,1} - \tau_{1,1,2} \tag{5}$$

$$\omega_{2,2} = \textstyle\sum_u \binom{d_2(u)}{2} - 3\tau_{2,2,2} - \tau_{1,2,2} \tag{6}$$

$$\omega_{1,2} = \textstyle\sum_u d_1(u)d_2(u) - 2\tau_{1,1,2} - 2\tau_{1,2,2} \tag{7}$$

Now let η_1 and η_2 be the counts of the 1-edge, 3-node induced subgraphs, where again the tie strength of the edge is given by the subscript (see the first row of Table 6). Denote the total number of weak and strong ties by $m_s = \frac{1}{2}\sum_u d_s(u)$, $s = 1, 2$, and the total number of nodes by n. Then the total number of (non-induced) 1-edge, 3-node subgraphs with tie strength s is then $m_s(n-2)$. Induced 1-edge, 3-node subgraph are given by the non-induced counts minus the 2- and 3-edge induced counts discussed above:

$$\eta_1 = m_1(n-2) - 2\omega_{1,1} - \omega_{1,2} - 3\tau_{1,1,1} - 2\tau_{1,1,2} - \tau_{1,2,2} \tag{8}$$

$$\eta_2 = m_2(n-2) - 2\omega_{2,2} - \omega_{1,2} - 3\tau_{2,2,2} - 2\tau_{1,2,2} - \tau_{1,1,2} \tag{9}$$

Finally, let ϕ be the number of empty 3-node induced subgraphs of the projected graph (the top left of Table 6). The number of subsets of 3 nodes minus all other induced 3-node subgraphs gives the value of ϕ:

$$\phi = \tbinom{n}{3} - \textstyle\sum_{s=1}^{2} \eta_s - \sum_{1 \le i,j \le 2} \omega_{i,j} - \sum_{1 \le i \le j \le k \le 2} \tau_{i,j,k}. \tag{10}$$

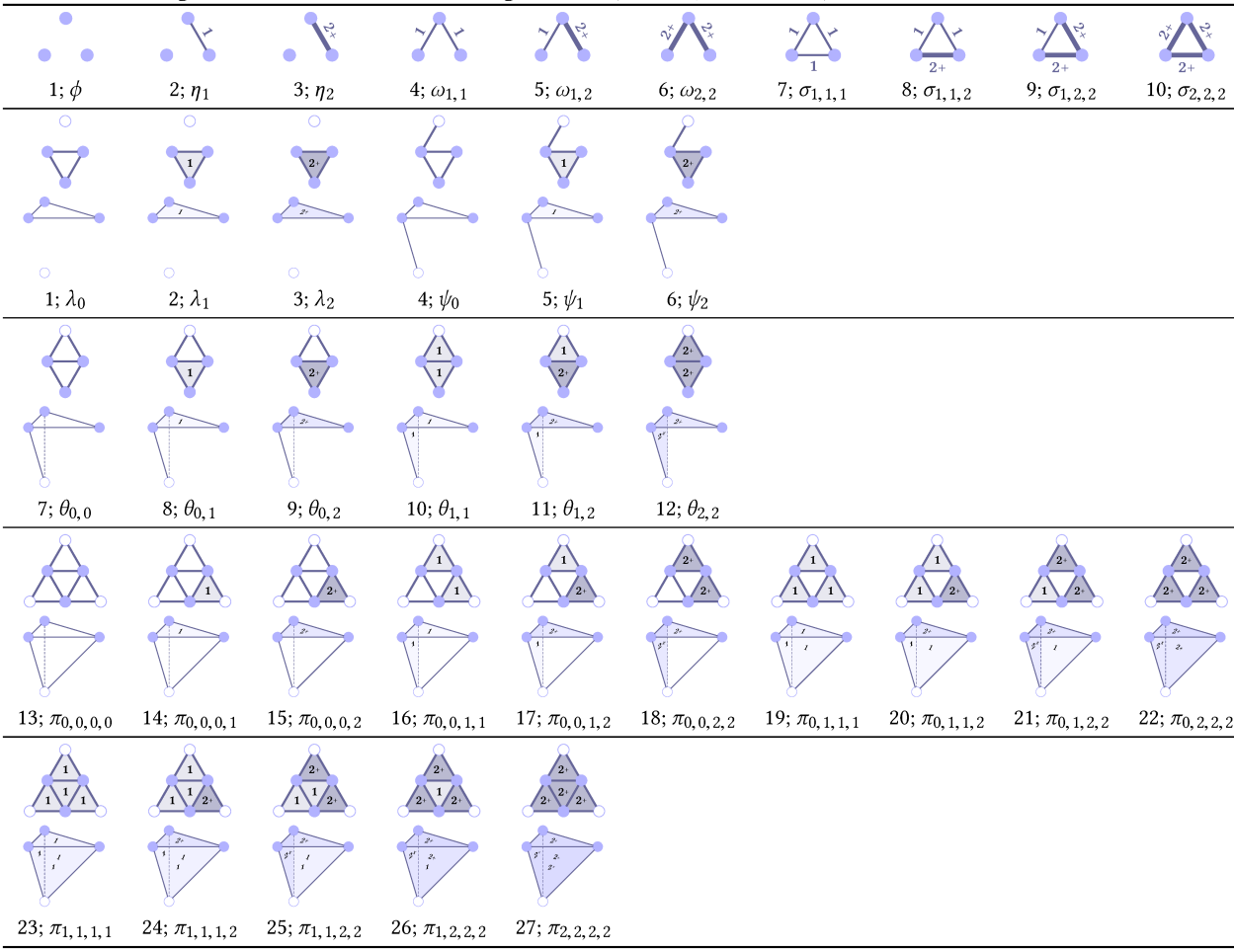
D.2 Counting for 4-node configurations

Now we describe how we compute the simplicial closure probabilities conditioned on the 27 subgraph configurations on 4 nodes in Fig. 5 (these are the 4-node configurations in the second through fifth row blocks of Table 6). Recall that the simplicial tie strength of a triangle is (i) *open* if the three nodes form an open triangle; (ii) *weak* if the three nodes have jointly appeared in exactly one simplex; or (iii) *strong* if the three nodes have jointly appeared in at least two simplices. We use subscripts 0, 1, and 2 to denote these three strengths.

There are 15 total 4-node, 6-edge tetrahedral subgraph configurations. Each configuration corresponds to a non-decreasing 4-tuple of the simplicial tie strengths of the four triangles in the configuration. We denote the sum of open and closed tetrahedral counts by $\rho_{i,j,k,l}$, where i, j, k, and l denote the simplicial tie strengths, and the open tetrahedral counts by $\pi_{i,j,k,l}$ ($0 \le i \le j \le k \le l \le 2$; the 15 configurations in the bottom two row blocks of Table 6). We may count both $\rho_{i,j,k,l}$ and $\pi_{i,j,k,l}$ by enumerating 4-cliques using, e.g., the Chiba and Nishizeki algorithm [14] and checking if each 4-clique is closed or open by examining the simplex-node adjacency matrix.

Next, we consider counts of the six 4-node, 5-edge subgraph configurations $\theta_{i,j}$, where each configuration is given by a non-decreasing pair of simplicial tie strengths for the two triangles in the configuration (the third row block of Table 6). Each instance of this configuration consists of two triangles sharing one edge. We first use a fast triangle enumeration algorithm to compute matrices $Y^{(s)}$, $s \in \{0, 1, 2\}$, where $Y^{(s)}_{uv}$ is the number of triangles with simplicial tie strength s containing nodes u and v. The counts of the non-induced configuration, which we denote by $\theta'_{i,j}$, are then

Table 6: The 10 open 3-node configurations analyzed in Fig. 4 and the 27 open 4-node configurations analyzed in Fig. 5. We illustrate each 4-node configuration with a projection onto two dimensions (top—the unfilled circle represents the same node) as well as a tetrahedral three-dimensional perspective figure (bottom). For each configuration, we list (i) the number of the configuration, which is referenced in Section 3.3 from the main text and (ii) our notation for the count of the number of instances of the configuration. For 3-ndoe configurations, the subscripts 1 and 2 denote weak and strong simplicial ties, and for 4-node configurations, the subscripts 0, 1, and 2 denote open, weak, and strong simplicial ties. We also use $\tau_{i,j,k}$ to denote the sum of counts of open and closed 3-node, triangles ($1 \le i \le j \le k \le 2$) and $\rho_{i,j,k,l}$ to denote the sum of counts of open and closed 4-node, 6-edge tetrahedral wireframe configurations ($0 \le i \le j \le k \le l \le 2$).



given by:

$$\theta'_{s,s} = \sum_{(u,v)} \binom{Y^{(s)}_{uv}}{2}, \quad s = 0, 1, 2 \tag{11}$$

$$\theta'_{i,j} = \sum_{(u,v)} Y^{(i)}_{uv} Y^{(j)}_{uv}, \quad 0 \le i < j \le 2. \tag{12}$$

The summations are over the edges (u, v) in the projected graph. Each non-induced instance of these subgraph configurations may correspond to a 6-edge tetrahedral configuration, and we need to adjust for these cases. Each (open or closed) 6-edge tetrahedron count $\rho_{i,j,k,l}$ contributes to the non-induced counts $\theta'_{i,j}$, $\theta'_{i,k}$, $\theta'_{i,l}$, $\theta'_{j,k}$, $\theta'_{j,l}$, and $\theta'_{k,l}$. To get the count $\theta_{i,j}$, we subtract the portion of $\theta'_{i,j}$ coming from the tetrahedra. Denote the set of valid 4-tuples of indices for the counts $\rho_{i,j,k,l}$ by $\mathcal{S}$. Formally, $\mathcal{S} = \{(i, j, k, l) \mid 0 \le i \le j \le k \le l \le 2\}$. Then $\theta_{i,j}$ is given by

$$\begin{aligned}\theta_{i,j} = \theta'_{i,j} \\ - \textstyle\sum_{k,l\,:\,(i,j,k,l)\in\mathcal{S}} \rho_{i,j,k,l} - \sum_{k,l\,:\,(i,k,j,l)\in\mathcal{S}} \rho_{i,k,j,l} \\ - \textstyle\sum_{k,l\,:\,(i,k,l,j)\in\mathcal{S}} \rho_{i,k,l,j} - \sum_{k,l\,:\,(k,i,j,l)\in\mathcal{S}} \rho_{k,i,j,l} \\ - \textstyle\sum_{k,l\,:\,(k,i,l,j)\in\mathcal{S}} \rho_{k,i,l,j} - \sum_{k,l\,:\,(k,l,i,j)\in\mathcal{S}} \rho_{k,l,i,j}.\end{aligned} \tag{13}$$

Next, we show how to count 4-node, 4-edge subgraph configurations that contain one triangle. There are three such configurations, corresponding to the three possible simplicial ties in the triangle, and we denote the counts by ψ_s, $s \in \{0, 1, 2\}$ (the three right-most configurations in the second row block of Table 6). We again compute non-induced counts and then subtract the induced counts of subgraphs with more edges, for which we showed how to compute above. Some additional notation will be helpful for these

counts. Let $\mathcal{T}_s$ be the set of triangles with simplicial tie strength $s \in \{0, 1, 2\}$, and let a_s and b_s count how many times triangles with a particular tie strength appear in 5-edge configuration patterns and 6-edge configuration patterns:

$$a_s = \sum_{0 \le i \le j \le 2} (\text{Ind}[i = s] + \text{Ind}[j = s])\theta_{i,j} \tag{14}$$

$$b_s = \sum_{(i,j,k,l) \in \mathcal{S}} (\text{Ind}[i = s] + \text{Ind}[j = s] + \text{Ind}[k = s] + \text{Ind}[l = s])\rho_{i,j,k,l}. \tag{15}$$

Consider a fixed triangle (u, v, w) with simplicial tie strength s. We would like to count the number of times this triangle appears in a 4-node, 4-edge subgraph configuration. Each neighbor of each of the three nodes in the triangle is either (i) the neighbor of just one node in the triangle (ii) the neighbor of exactly two nodes in the triangle, or (iii) the neighbor of all three nodes in the triangle. The first case corresponds to the induced subgraph in which we are interested, the second case to counts $\theta_{i,j}$, and the third case to counts $\rho_{i,j,k,l}$. By the inclusion-exclusion principle,

$$\psi_s = \sum_{(u,v,w) \in \mathcal{T}_s} (d_u + d_v + d_w - 6) - 2a_s - 3b_s, \tag{16}$$

where d is the degree vector of nodes in the unweighted projected graph.

Finally, we count the 4-node subgraph configuration consisting of a triangle and an isolated node (the three leftmost configurations in the second row block of Table 6). Again, we count three types of this configuration (λ_s, $s \in \{0, 1, 2\}$), one for each of the three simplicial tie strengths of the triangle. Every triangle appears in $(n - 3)$ non-induced subgraphs with an isolated node, so we only need to subtract induced subgraph counts with more edges. We already counted these above, so the counts λ_s are given by

$$\lambda_s = |\mathcal{T}_s|(n - 3) - \psi_s - a_s - b_s. \tag{17}$$

E SCORE FUNCTIONS FOR HIGHER-ORDER LINK PREDICTION AND EXAMPLE PREDICTIONS

We derive algorithms for higher-order link prediction, which fall into four broad categories for determining the score $s(i, j, k)$ of a triple of nodes:

(1) $s(i, j, k)$ depends only on the weights of the edges (i, j), (i, k), and (j, k) in the projected graph
(2) $s(i, j, k)$ is based on the local neighborhood features in the projected graph such as the common neighbors of nodes i, j, and k;
(3) $s(i, j, k)$ comes from a random-walk-based similarity score
(4) $s(i, j, k)$ is a learned logistic regression model in a feature-based supervised learning setting.

Several of these score functions are generalizations of traditional approaches for dyadic link prediction [34] to account for higher-order structure.

Here we introduce some notation for this section. We denote the set of simplices that node u appears in by $R(u)$; formally, $R(u) = \{S_i \mid u \in S_i\}$. The (weighted) projected graph of a dataset is the graph on node set V, where the weight of edge (u, v) is the number of simplices containing both u and v. In other words, the $|V| \times |V|$ weighted adjacency matrix W of the projected graph is defined by

$$W_{uv} = \begin{cases} |R(u) \cap R(v)| & u \neq v \\ 0 & u = v \end{cases} \tag{18}$$

Sometimes, we will only need to consider unweighted version of the projected graph, which is encoded by the adjacency matrix A with entries $A_{uv} = \min(W_{uv}, 1)$. Finally, we denote the neighbors of a node u in the projected graph by $N(u) = \{v \in V \mid W_{uv} > 0\}$.

E.1 Weights in the projected graph

We use three score functions based on the weights of the pair-wise edges in the subgraph induced by nodes i, j, and k. The motivation for these methods is that weight-based tie strength positively correlates with simplicial closure probability in an aggregate sense (see Fig. 4). Therefore, larger weights amongst the edges between nodes i, j, and k should yield larger scores. To this end, we use the the *harmonic mean*,

$$s(i, j, k) = 3/(W_{ij}^{-1} + W_{ik}^{-1} + W_{jk}^{-1}), \tag{19}$$

the *geometric mean*,

$$s(i, j, k) = \left(W_{ij} W_{ik} W_{jk}\right)^{1/3}, \tag{20}$$

and *arithmetic mean*,

$$s(i, j, k) = (W_{ij} + W_{ik} + W_{jk})/3, \tag{21}$$

as score functions. As discussed in the main text, these functions are all special cases of the generalized mean function (Eq. (3)).

E.2 Local neighborhood features

The next set of score functions use local neighborhood features such as common neighbors of a triple of nodes. The reasoning here is that common neighborhood structure amongst a triple of nodes are positive indicators of association of the nodes; in fact, these score functions are generalizations of traditional methods used in dyadic link prediction [34]. The *3-way common neighbors* score function for a triple of nodes i, j, and k is the number of nodes that have appeared in at least one simplex with each of the three nodes in the candidate set:

$$s(i, j, k) = |N(i) \cap N(j) \cap N(k)|, \tag{22}$$

where again $N(x)$ is the set of neighbors of node x in the projected graph.

The *3-way Jaccard coefficient* score normalizes the number of common neighbors by the total number of neighbors of the three candidate nodes:

$$s(i, j, k) = \frac{|N(i) \cap N(j) \cap N(k)|}{|N(i) \cup N(j) \cup N(k)|}. \tag{23}$$

This score function has been used as a general multi-way similarity measurement for binary vectors [25], but has not been employed for a link prediction task until now.

Adamic and Adar proposed log-scaled normalization for features of common neighbors between two nodes [1]. Here we adapt this to a *3-way Adamic-Adar* score that performs the same normalization over the common neighbors of 3 nodes:

$$s(i, j, k) = \sum_{l \in N(i) \cap N(j) \cap N(k)} \frac{1}{\log |N(l)|}. \tag{24}$$

Prior studies on the evolution of co-authorship have suggested preferential attachment (in terms of degree in the co-authorship network) as a mechanism for dyadic link formation [7, 42]. We use two scores based on a preferential attachment model of link formation. The first is *projected graph degree based preferential attachment*,

$$s(i, j, k) = |N(i)| \cdot |N(j)| \cdot |N(k)|, \tag{25}$$

and the second is *simplicial degree based preferential attachment*,

$$s(i, j, k) = |R(i)| \cdot |R(j)| \cdot |R(k)|. \tag{26}$$

E.3 Paths and random walks

The next set of scores are path-based metrics that ascribe higher scores when there are more paths in the projected graph between a candidate triple of nodes. Recall that A and W are the unweighted and weighted adjacency matrices for the projected graph of a dataset.

The Katz score between two nodes is the sum of geometrically damped length-l paths between two nodes [28]. Katz scores have been used as a criterion for predicting dyadic links [34, 64]. Formally, the Katz score between two nodes i and j in the unweighted projected graph is $\sum_{l=1}^{\infty} \beta^l A_{ij}^l$, where β is the damping parameter and A_{ij}^l counts the number of length-l paths between i and j. All pairwise Katz scores can be computed in matrix form as:

$$K^{(u)} = (I - \beta A)^{-1} - I. \tag{27}$$

In order to guarantee that the weighted sum of length-l path lengths converges, we require that $\beta < 1/\sigma_1(A)$, the principal singular value of A (this guarantees that $I - \beta A$ is nonsingular). We chose $\beta = \frac{1}{4\sigma_1(A)}$ in our experiments.

We can also use paths in the original (weighted) projected graph, where W_{ij}^l is the number of length-l paths between i and j if we

interpret the integer weights in W to be parallel edges. This leads to the weighted pairwise Katz scores

$$K^{(w)} = (I - \beta W)^{-1} - I. \tag{28}$$

Again, β must be less than $1/\sigma_1(W)$ to guarantee that $(I - \beta W)$ is nonsingular, and we choose $\beta = \frac{1}{4\sigma_1(W)}$ in our experiments.

Given the pairwise Katz scores, we define score functions for triplets, based on the pairwise Katz scores. The *3-way Katz score* is

$$s(i,j,k) = K^{(u)}_{ij} + K^{(u)}_{ik} + K^{(u)}_{jk}, \tag{29}$$

and the *weighted 3-way Katz score is*

$$s(i,j,k) = K^{(w)}_{ij} + K^{(w)}_{ik} + K^{(w)}_{jk}, \tag{30}$$

For many of our datasets, storing the K matrices in a dense format requires too much memory. In these cases, we use the Krylov subspace method GMRES [57] (with tolerance 10^{-3}) to solve the linear systems

$$(I - \beta A)k_j = e_j, \quad j = 1, \ldots, |V|, \tag{31}$$

where e_j is the jth standard basis vector. After computing k_j, we store only the entries of the jth column of K corresponding to the sparsity pattern of the jth column of A. These are the only entries of K needed for computing the scores in Eq. (29).

The personalized PageRank score is another path-based score used in dyadic link prediction [5, 34]. Personalized PageRank is based on the random walk underlying the classical PageRank ranking system for web pages [46]. More specifically, consider a Markov chain, where at each step, with probability $0 < \alpha < 1$, the chain transitions according to a random walk in a graph, and with probability $1 - \alpha$ transitions to node i. The personalized PageRank score of node j with respect to node i is then the stationary probability of the state j for the Markov chain. The pairwise personalized PageRank scores are given by the matrix

$$F^{(u)} = (1 - \alpha)(I - \alpha A D^{-1})^{-1}, \tag{32}$$

where $F^{(u)}_{ji}$ is the personalized PageRank score of j with respect to node i. Here D is the diagonal degree matrix, $D_{jj} = \sum_i A_{ij}$. We can again provide an analog for the weighted case:

$$F^{(w)} = (1 - \alpha)(I - \alpha W D_W^{-1})^{-1}, \tag{33}$$

where $[D_W]_{jj} = \sum_i W_{ij}$ is the weighted diagonal degree matrix.

As we did with the Katz scores, we construct three-way scores from the pairwise personalized PageRank scores. The *3-way personalized PageRank score* is

$$s(i,j,k) = F^{(u)}_{ij} + F^{(u)}_{ji} + F^{(u)}_{ik} + F^{(u)}_{ki} + F^{(u)}_{jk} + F^{(u)}_{kj}, \tag{34}$$

and the *weighted 3-way personalized PageRank score* is

$$s(i,j,k) = F^{(w)}_{ij} + F^{(w)}_{ji} + F^{(w)}_{ik} + F^{(w)}_{ki} + F^{(w)}_{jk} + F^{(w)}_{kj}. \tag{35}$$

(Unlike the Katz score matrices K, the personalized PageRank matrices are not symmetric, so we account for both directions of the edges.)

We also use a recent generalization of the personalized PageRank score for abstract simplicial complexes, based on tools from algebraic topology [27]. Here, we describe the computations necessary for these scores, assuming a basic knowledge of algebraic topology. We provide a brief derivation of the method and the necessary topology background in the next section of the supporting information.

We consider the abstract simplicial complex defined by the union of the set of closed triangles T, the set of edges E, and the set of vertices V. We orient the edges and triangles so that (i,j) for $i < j$ corresponds to an edge $\{i,j\}$ and (i,j,k) for $i < j < k$ corresponds to a closed triangle $\{i,j,k\}$. Following the ideas of Horn et al. [27], we define the normalized combinatorial Hodge Laplacian as

$$\hat{\Delta} = (GD^{-1}G^T + C^T C)M^{-1}, \tag{36}$$

where the "gradient operator" G is a $|E| \times |V|$ matrix defined by

$$G_{(i,j),x} = \begin{cases} 1 & x = j \\ -1 & x = i \\ 0 & \text{otherwise,} \end{cases} \tag{37}$$

the "curl operator" C is a $|T| \times |E|$ matrix defined by

$$C_{(i,j,k),(x,y)} = \begin{cases} 1 & (x,y) = (i,j) \text{ or } (x,y) = (j,k) \\ -1 & (x,y) = (i,k) \\ 0 & \text{otherwise,} \end{cases} \tag{38}$$

D is a diagonal matrix defined by

$$D_{xx} = \textstyle\sum_{(i,j)} |G_{(i,j),x}|, \tag{39}$$

and M is a diagonal matrix defined by

$$M_{(x,y),(x,y)} = 2 + \textstyle\sum_{(i,j,k)} |C_{(i,j,k),(x,y)}|. \tag{40}$$

The matrix $P = \frac{1}{2}(I - \hat{\Delta})$ defines a Markov-like operator (see next section). The simplicial PageRank scores (defined on each pair of edges) can thus be defined analogously to the standard PageRank:

$$S = (I - \alpha P)^{-1}(1 - \alpha). \tag{41}$$

Here, the matrix S defines pairwise scores between *edges*, and we construct a score function on triples of nodes by taking the sum of pairwise scores. The *3-way simplicial personalized PageRank score* is

$$\begin{aligned} s(i,j,k) = {} & |S_{(i,j),(j,k)}| + |S_{(j,k),(i,j)}| \\ & + |S_{(i,j),(i,k)}| + |S_{(i,k),(i,j)}| \\ & + |S_{(j,k),(i,k)}| + |S_{(j,k),(i,k)}|. \end{aligned} \tag{42}$$

We may further decompose the pairwise scores into the gradient, harmonic, and curl components given by the Hodge decomposition. Computationally, we solve the least squares problems

$$\min_X \|GX - S\|_F, \qquad \min_Y \|C^T Y - S\|_F \tag{43}$$

using the iterative method LSQR [47] (with tolerances 10^{-3}) on each column. Given the minimizers X^* and Y^* of Eq. (43), the components of the Hodge decomposition are

$$S_{\text{grad}} = GX^* \tag{44}$$

$$S_{\text{curl}} = C^T Y^* \tag{45}$$

$$S_{\text{harm}} = S - S_{\text{grad}} - S_{\text{curl}}. \tag{46}$$

Each of S_{grad}, S_{curl}, and S_{harm} defines pairwise scores between edges, and we construct score functions on triples of nodes in the same way as in Eq. (42).

Table 7: Open triangle closure prediction performance based on score functions from the Hodge decomposition of the simplicial personalized PageRank vector.

Dataset	Rand.	combined	gradient	harmonic	curl
coauth-MAG-History	7.16e-04	1.35	1.25	1.13	1.27
music-rap-genius	6.82e-04	1.39	1.44	1.40	1.47
tags-math-sx	1.08e-03	1.86	0.73	0.66	0.74
tags-ask-ubuntu	1.08e-03	1.19	0.61	0.59	0.71
threads-ask-ubuntu	1.31e-04	0.61	0.58	0.61	4.59
NDC-substances	1.17e-03	1.86	0.63	0.72	0.60
NDC-classes	6.72e-03	2.45	1.37	0.83	1.74
DAWN	8.47e-03	1.55	0.59	0.60	0.65
congress-committees	6.99e-04	2.13	1.22	1.13	1.63
email-Enron	1.40e-02	2.02	2.90	1.98	2.46
email-Eu	5.34e-03	1.26	1.28	0.82	1.63
contact-high-school	2.47e-03	0.78	0.99	1.68	2.38
contact-primary-school	2.59e-03	0.93	1.45	1.84	3.26

We report the performance results in Table 7 (analogous to those in Table 2) for the datasets that were small enough on which computing the Hodge decomposition was computationally feasible. We observe that the components from the Hodge decomposition can provide substantially better results than the "combined" simplicial PageRank score reported in the main text in Table 2. However, no component consistently out-performs the others.

E.4 Supervised learning

Finally, we used a supervised machine learning approach that learns the appropriate score function given features of the open triangle. To this end, we further divide the training data into a sub-training set (simplices appearing in the first 60% of the entire dataset) and a validation set (simplices appearing between the 60th and 80th percentile of the time spanned by the entire dataset). We trained an ℓ_2-regularized logistic regression model[7] for predicting closure on the validation set using features of open structures in the sub-training set. The features for each open triangle (i, j, k) were

(1) the number of simplices containing pairs of nodes i and j, i and k, and j and k;
(2) the degree of nodes i, j, and k in the projected graph: $|N(i)|$, $|N(j)|$, and $|N(k)|$;
(3) the number of simplices containing nodes i, j, and k: $|R(i)|$, $|R(j)|$, and $|R(k)|$;
(4) the number of common neighbors in the projected graph of nodes i and j, i and k, and j and k: $|N(i) \cap N(j)|$, $|N(i) \cap N(k)|$, and $|N(j) \cap N(k)|$;
(5) the number of common neighbors of all three nodes i, j, and k in the projected graph: $|N(i) \cap N(j) \cap N(k)|$
(6) the log of the features in Items 1 to 3 and the log of the sum of 1 and the feature value for the features in Items 4 and 5.

After learning the model, we predicted on the test set using the same features computed on the entire training set (first 80% of the dataset).

[7] Using the scikit learn library [51]: http://scikit-learn.org/

Table 8: Top 25 predictions from the 3-way Adamic-Adar algorithm for open triangles to simplicially close in the DAWN dataset. An "X" marks open triangles that actually simplicially close in the final 20% of the time spanned by the dataset. Four of the top 25 predictions simplicially close.

1		methyldopa; gentamicin; proton pump inhibitors
2	X	norepinephrine; chlormezanone; proton pump inhibitors
3		ranitidine; gentamicin; proton pump inhibitors
4		dihydroergotamine; methyldopa; asa/butalbital/caffeine/codeine
5		ranitidine; gentamicin; levodopa
6		praziquantel; diazepam; alfentanil
7		asa/caffeine/dihydrocodeine; praziquantel; proton pump inhibitors
8		chloral hydrate; tobramycin; sumatriptan
9		oxybutynin; gentamicin; tobramycin
10		asa/caffeine/dihydrocodeine; norepinephrine; sumatriptan
11		ampicillin; chlormezanone; proton pump inhibitors
12		bepridil; diazepam; alfentanil
13		colestipol; oxybutynin; proton pump inhibitors
14	X	nadolol; benazepril; proton pump inhibitors
15		thalidomide; amiloride; maprotiline
16	X	nadolol; lamivudine-zidovudine; proton pump inhibitors
17		chloral hydrate; verapamil; methyldopa
18		chlorzoxazone; benazepril; proton pump inhibitors
19		heparin; asa/caffeine/dihydrocodeine; proton pump inhibitors
20		oxcarbazepine; norepinephrine; proton pump inhibitors
21		dihydroergotamine; tobramycin; alfentanil
22		maprotiline; norepinephrine; proton pump inhibitors
23		oxybutynin; methyldopa; dihydroergotamine
24		heparin; dihydroergotamine; proton pump inhibitors
25	X	ampicillin; methyldopa; diazepam

E.5 Example predictions

Finally, Table 8 provides a concrete example of the predictions from our framework—specifically, the top 25 predictions of the Adamic-Adar score function on the DAWN dataset. In this dataset, fewer than one in a hundred open triangles in the training set simplicially close in the test set, but 4 of the top 25 predictions from this score function simplicially close. Three of the correct predictions relate to novel combinations with proton pump inhibitors.

F ADDITIONAL TOPOLOGY BACKGROUND AND SIMPLICIAL PAGERANK

In the following we provide a brief summary of simiplicial complexes and notions from algebraic topology underpinning some of the results presented in the manuscript. The presentation here is deliberately simple and geared towards conveying intuition rather than exposing all the mathematical details. We refer to Hatcher's book [24] for rigorous details as well as Lim's survey [35] for readable introductions to the topic.

Given a finite set of nodes V, we define a k-node simplex S_i as a subset of k nodes. In more standard topological parlance, V would be called the set of points, and a subset of $k+1$ of those points would be denoted a k-simplex. We deviate from this standard here to emphasize the relationships to graphs, which might be more familiar for a reader not acquainted with (algebraic) topology. A *face* of a k-node simplex S_i is a $(k-1)$-node simplex $S_j \subset S_i$ that is a proper subset of S_i. An *abstract simplicial complex* K is now a finite collection of simplices $\{S_i\}$ that is closed with respect to inclusion of faces, i.e., if the k-node simplex S_i is part of K, then all of its faces are also part of K. Simplicial complexes may intuitively be regarded as generalizations of graphs, in which not only binary but also higher-order relationships between the vertices are allowed. More concretely, a graph is a finite collection of 1-node simplices (the vertices) and 2-node simplices (the edges). Moreover, as an edge can of course only exist between two nodes that are present in the graph, the closure condition for the faces is fulfilled for a graph.

Simplicial complexes may also be seen as a special form of hypergraph, in which a hyperedge can only be present if all the possible subsets of the hyperedge are also present in the graph. While this restricts the flexibility of simplicial complexes as a modeling tool compared to generic hypergraphs, there is an intimate relationships between simplicial complexes and algebraic topology. Indeed, simplical complexes carry additional algebraic structure that can be exploited in applications such as ours. Also, observe that the data we consider in this paper is set-valued, i.e., we consider sets of (interacting) nodes appearing over time. Simplicial complexes are a natural modeling framework in this context, as observing a set of nodes S_i together implies that all the subsets of S_i have been observed together, too.

For the rest of this section, we consider the abstract simplicial complex whose largest simplices are closed triangles in the dataset. Formally, the complex is the union of vertices, edges and closed triangles. Thus, every element of our abstract simplicial complex is a 1-node, 2-node, or 3-node simplex.

The graph Helmholtzian. In close analogy to graphs, one can define Laplacian operators for simplicial complexes (the so-called Hodge Laplacian). For the special case of our abstract simplicial complex induced by triangles, this higher-order Laplacian is called the *graph Helmholtzian*.

We write the graph Helmholtzian as a linear operator on the space of alternating functions on the edges of the projected graph, following the presentation and notation of Lim [35]. Let $H(V)$ be the Hilbert space of function on the vertices of our data, where the inner product is

$$\langle f, g\rangle_V = \sum_{i\in V} f(i)g(i) \tag{47}$$

Let $H_\wedge(E)$ be the Hilbert space of alternating functions on the edges of the projected graph, where the inner product is

$$\langle X, Y\rangle_E = \sum_{(i,j)\in E} X(i,j)Y(i,j). \tag{48}$$

Here, E is the set of edges in the projected graph and alternating means that $X(i,j) = -X(i,j)$. Note that we sum over each edge only once, i.e., we may consider the sum to be over all indices with $i < j$. Further, $X(i,j) = 0$ for all $(i,j) \notin E$.

We similarly define $H_\wedge(T)$ to be the Hilbert space of alternating functions on the set of closed triangles T, where the inner product is

$$\langle \Phi, \Psi\rangle_T = \sum_{(i,j,k)\in C} \Phi(i,j,k)\Psi(i,j,k). \tag{49}$$

Again the sum is taken, such that each cycle is counted exactly once. In the space of triangles, the fact that the functions we consider are alternating means that

$$\begin{aligned}\Phi(i,j,k) &= \Phi(k,i,j) = \Phi(j,k,i)\\ &= -\Phi(i,k,j) = -\Phi(j,i,k) = -\Phi(k,j,i)\end{aligned}$$

Further, $\Phi(i,j,k) = 0$ for all $(i,j,k) \notin C$.

We define the operators $\mathrm{grad}\colon H(V) \to H_\wedge(E)$ and $\mathrm{curl}\colon H_\wedge(E) \to H_\wedge(T)$ as follows:

$$(\mathrm{grad}\, f)(i,j) = f(j) - f(i) \tag{50}$$

$$(\mathrm{curl}\, X)(i,j,k) = X(i,j) + X(j,k) + X(k,i) \tag{51}$$

These operators are indeed the discrete equivalents of the gradient and curl operators known from continuous vector calculus, thus justifying our naming scheme. For more details refer to Lim [35].

In the language of algebraic topology, the above operators are co-boundary operators on our simplicial complex. One can check that the adjoints of these linear operators are $\mathrm{grad}^*\colon H_\wedge(E) \to H(V)$ and $\mathrm{curl}^*\colon H_\wedge(T) \to H_\wedge(E)$, defined via

$$(\mathrm{grad}^* X)(i) = -\sum_{j=1}^{n} X(i,j) \tag{52}$$

$$(\mathrm{curl}^* \Phi)(i,j) = \sum_{k=1}^{n} \Phi(i,j,k) \tag{53}$$

Sometimes the negative of the grad^* is called the *divergence* operator. The Hodge Laplacian linear operator on $H_\wedge(E)$, which we denote by $\Delta\colon H_\wedge(E) \to H_\wedge(E)$, is called the graph Helmholtzian [35]:

$$\Delta = \mathrm{grad}\ \mathrm{grad}^* + \mathrm{curl}^* \mathrm{curl}. \tag{54}$$

Using these definitions, it can be seen that the standard graph Laplacian $\mathcal{L}$ can be written as:

$$\mathcal{L} = \mathrm{grad}^* \mathrm{grad}. \tag{55}$$

*The Hodge decomposition on edge flows.*The space of alternating function on edges, $H_\wedge(E)$, is sometimes called the space of *edge flows*, as an element $X \in H_\wedge(E)$ induces a flow on the graph. The space $H_\wedge(E)$ can be decomposed into three orthogonal parts, each of which with a special meaning in terms of flows. The *Hodge decomposition* provides this decomposition:

$$H_\wedge(E) = \mathrm{im}(\mathrm{grad}) \oplus \mathrm{null}(\Delta) \oplus \mathrm{im}(\mathrm{curl}^*) \tag{56}$$

In this decomposition, im(grad) is called the cut space, or the gradient component of an edge flow [10]. It consists of all the flows which have no cyclic component, i.e., their sum along any cyclic

path in the graph is zero, taking into account the orientations of the edges. Following Eq. (50), any such flow can be obtained by defining a scalar potential on each node, and the edge flows are the difference of the scalar potentials at its two endpoints. Second, im(curl*) consists of all flows that can be composed out of local circulations along any 3-node simplex, i.e., a circulation around a closed triangle. Third, null(Δ) is the space of harmonic flows, which correspond to those flows which are locally circulation free in that they cannot be composed from curl flows. However, harmonic flows are not expressible as gradients either, in that they contain a global circulatory component and thus do not sum to zero around every cyclic path. The name harmonic component derives from the fact, that the Helmholtzian is in fact the discrete analog of the Helmholtz or vector Laplacian operator in the continuous domain. The dimension of null(Δ) is a *Betti number* of the simplicial complex. In our case, it corresponds to the number of "holes" in the simplicial complex.

Extending PageRank to simplicial complexes. We now give an overview of the ideas of Horn et al. for defining a suitable notion of PageRank for simplicial complexes [27]. The central idea is to relate the notion of PageRank on graphs to a normalization of the graph Laplacian operator. We then provide a normalization for the graph Helmholtzian to derive a PageRank formulation for simplicial complexes.

Recall that the PageRank operator is defined via a Markov operator P, acting on $H(V)$. More specifically, the Markov operator is given by a random walk on the graph, which can be written as

$$P = I - \text{grad}^*\text{grad}D^{-1} = I - \mathcal{L}D^{-1}. \tag{57}$$

Consider the vector p_t, describing the probability of a random walker being present at any of the nodes at time t. Then, the random walk on the graph can be written as $p_{t+1} = Pp_t$.

The personalized PageRank scores v with respect to node i are solutions

$$(I - \alpha P)v = e_i(1-\alpha), \tag{58}$$

where $e_i \in H(V)$ takes value 1 at node i and 0 elsewhere.

The idea of Horn et al. is to normalize the graph Helmholtzian Δ to a linear operator $\hat{\Delta}$ which can also be interpreted from a Markov chain viewpoint. Specifically, they introduce the following normalized Helmholtzian

$$\hat{\Delta} = (\text{grad}\, D^{-1}\text{grad}^* + \text{curl}^*\text{curl})M^{-1}, \tag{59}$$

where $D^{-1}: H(V) \to H(V)$ is a diagonal node scaling operator,

$$(D^{-1}f)(i) = \frac{1}{d_i}f(i), \quad d_i = |\{j : \{i,j\} \in E\}|, \tag{60}$$

and $M^{-1}: H_\wedge(E) \to H_\wedge(E)$ is a diagonal edge scaling operator,

$$(M^{-1}X)(i,j) = \frac{1}{m_{i,j}}X(i,j) \tag{61}$$

$$m_{i,j} = 2 + |\{k : \{i,j,k\} \in C\}| =: 2 + d_{i,j}. \tag{62}$$

We now show how $\hat{\Delta}$ maps elements of $H_\wedge(E)$:

$$(\text{curl}^*\text{curl}\, X)(i,j) \tag{63}$$
$$= \textstyle\sum_{k:\{i,j,k\}\in C}(\text{curl}X)(i,j,k) \tag{64}$$
$$= \textstyle\sum_{k:\{i,j,k\}\in C} X(i,j) + X(j,k) + X(k,i) \tag{65}$$
$$= d_{i,j}X(i,j) + \textstyle\sum_{k:\{i,j,k\}\in C}[X(j,k) + X(k,i)] \tag{66}$$

and

$$(\text{grad}\, D^{-1}\text{grad}^*X)(i,j) \tag{67}$$
$$= (D^{-1}\text{grad}^*X)(j) - (D^{-1}\text{grad}^*X)(i) \tag{68}$$
$$= \tfrac{1}{d_j}(\text{grad}^*X)(j) - \tfrac{1}{d_i}(\text{grad}^*X)(i) \tag{69}$$
$$= \tfrac{1}{d_i}\textstyle\sum_{k:\{i,k\}\in E} X(i,k) - \tfrac{1}{d_j}\sum_{k:\{j,k\}\in E} X(j,k) \tag{70}$$

Let us now define the transpose operator $\hat{\Delta}^\top$ by putting everything together and including the normalization from M^{-1}.

$$(\hat{\Delta}^T X)(i,j) \tag{71}$$
$$= \tfrac{1}{2+d_{i,j}}[d_{i,j}X(i,j) + \textstyle\sum_{\{i,j,k\}\in C} X(j,k) + X(k,i) \tag{72}$$
$$+ \tfrac{1}{d_i}\textstyle\sum_{\{i,k\}\in E} X(i,k) - \tfrac{1}{d_j}\sum_{\{j,k\}\in E} X(j,k)], \tag{73}$$

Note that as (curl*curl) and (grad D^{-1}grad*) are symmetric this simply amounts to applying a row normalization instead of the column normalization by M, which will be useful in the sequel. We now define the operator $Q^T = \frac{1}{2}(I - \hat{\Delta})^T$,

$$(Q^T X)(i,j) \tag{74}$$
$$= \tfrac{1}{4+2d_{i,j}}[2X(i,j) - \textstyle\sum_{\{i,j,k\}\in C} X(j,k) + X(k,i) \tag{75}$$
$$- \tfrac{1}{d_i}\textstyle\sum_{\{i,k\}\in E} X(i,k) + \tfrac{1}{d_j}\sum_{\{j,k\}\in E} X(j,k)] \tag{76}$$
$$= \tfrac{1}{4+2d_{i,j}}[2X(i,j) + \textstyle\sum_{\{i,j,k\}\in C} X(k,j) + X(i,k) \tag{77}$$
$$+ \tfrac{1}{d_i}\textstyle\sum_{\{i,k\}\in E} X(k,i) + \tfrac{1}{d_j}\sum_{\{j,k\}\in E} X(j,k)]. \tag{78}$$

Here, in the second equality, we used the fact that X is an alternating function.

Let us know define the operator $\tilde{Q}^T$ in exactly this way, but now acting on the whole space of (non-alternating) edge functions. It is now easy to see that $\tilde{Q}^T$ is a row stochastic Markov operator, i.e., if $Y(i,j) = Y(j,i) = 1$ for all $\{i,j\} \in E$,

$$(\tilde{Q}^T Y)(i,j) \tag{79}$$
$$= \frac{2 + \sum_{k:\{i,j,k\}\in C} 2 + \frac{1}{d_i}\sum_{k:\{i,k\}\in E} 1 + \frac{1}{d_j}\sum_{k:\{j,k\}\in E} 1}{4 + 2d_{i,j}} \tag{80}$$
$$= \frac{2 + 2d_{i,j} + \frac{1}{d_i}d_i + \frac{1}{d_j}d_j}{4 + 2d_{i,j}} = 1. \tag{81}$$

Hence, its transpose $\tilde{Q}$ is column stochastic and corresponds to a Markov operator acting on the space of edge functions.

The corresponding equivalent Q in the space of alternating edge functions may thus be related to a random walk on the edge space, which justifies the use the above scheme in a personalized PageRank style. In particular, the solution v to the system

$$\frac{1}{2}(I - \alpha Q)v = e_i \tag{82}$$

defines an edge flow, signifying how important each edge is to the ith edge.