

# Algorithmic Tools for Understanding the Motif Structure of Networks

Tianyi Chen<sup>1</sup>, Brian Matejek<sup>2,3</sup>, Michael Mitzenmacher<sup>2</sup>, and Charalampos Tsourakakis<sup>1,2,4</sup>

<sup>1</sup> Boston University, Boston MA, USA

<sup>2</sup> Harvard University, Cambridge MA, USA

<sup>3</sup> Computer Science Laboratory, SRI International, Washington, DC

<sup>4</sup> ISI Foundation, Italy

**Abstract.** Motifs are small subgraph patterns that play a key role towards understanding the structure and the function of biological and social networks. The current *de facto* approach towards assessing the statistical significance of a motif  $\mathcal{M}$  relies on counting its occurrences across the network, and comparing that count to its expected count under some null generative model. This approach can be misleading due to *combinatorial artifacts*. That is, there may be a large count for a motif due to multiple copies sharing many vertices and edges connected to a subgraph, such as a clique, that completes the multiple copies of the motif.

In this work we introduce the novel concept of an  $(f, q)$ -spanning motif. A motif  $\mathcal{M}$  is  $(f, q)$ -spanning if there exists a  $q$ -fraction of the nodes that induces an  $f$ -fraction of the occurrences of  $\mathcal{M}$  in  $G$ . Intuitively, when  $f$  is close to 1, and  $q$  close to 0, most of the occurrences of  $\mathcal{M}$  are localized in a small set of nodes, and thus its statistical significance is likely to be due to a combinatorial artifact. We propose efficient heuristics for finding the maximum  $f$  for a given  $q$  and minimum  $q$  for a given  $f$  for which a motif is  $(f, q)$ -spanning and evaluate them on real-world datasets. Our methods successfully identify combinatorial artifacts that otherwise go undetected using the standard approach for assessing statistical significance.

Finally, we leverage the motif structure of a network to design MOTIFSCOPE, an algorithm that takes as input a graph and two motifs  $\mathcal{M}_1, \mathcal{M}_2$ , and finds subgraphs of the graph where  $\mathcal{M}_1, \mathcal{M}_2$  occur infrequently and frequently respectively. We show that a good selection of  $\mathcal{M}_1, \mathcal{M}_2$  allows us to find anomalies in large networks, including bipartite cliques in social graphs, and subgraphs rated with distrust in Bitcoin markets.

**Keywords:** motifs, graph mining, statistical significance, anomaly detection

## 1 Introduction

Network motifs, or small induced subgraph patterns, are known to play a key role in understanding the structure and function of various real-world networks,

especially biological [28, 40], and social networks [47]. For example the feed-forward loop (FFL) is one of the most significant subgraphs in the transcription network of the bacteria *Escherichia coli*. The FFL has three nodes corresponding to transcription factors. The transcription factor  $X$  regulates a second transcription factor  $Y$ , and together they bind the regulatory region of a target gene  $Z$ , jointly modulating its transcription rate [27]. In social networks, triangles ( $K_3$ s) are known to appear frequently despite the edge sparsity of the network [49]. Ugander, Backstrom, and Kleinberg [47] showed that on the other hand social networks have very few cycles of length 4 ( $C_4$ s). This sheer contrast in the counts of  $K_3$ s and  $C_4$ s relates to human nature. Specifically, friends of friends are typically friends themselves, thus introducing edges that create  $K_3$ s but remove  $C_4$ s [49]. An FFL and a  $C_4$  are shown in Figure 1(a).

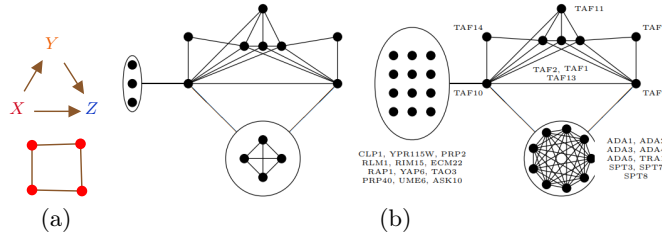


Fig. 1: (a) A feed-forward loop (FFL, top) and a  $C_4$  (bottom). (b) Figure source [17]: the subgraph  $\mathcal{M}$  on the left appears to be statistically significant in the network  $G$  on the right due to the presence of a large independent set, and a large clique in  $G$ . The independent set creates  $\binom{12}{3}$  stars with three leaves, while the large clique creates  $\binom{9}{4}$  smaller cliques of order 4, resulting in a total count of  $\binom{12}{3} \times \binom{9}{4}$  occurrences, leading to the misleading conclusion that  $\mathcal{M}$  is a statistically significant motif. We refer to this phenomenon as a combinatorial artifact, see also [32, 17].

The *de facto* current approach towards assessing the statistical significance of a motif  $\mathcal{M}$  involves two steps: (i) counting the occurrences of  $\mathcal{M}$  in the input graph, and (ii) comparing that count to the expected number of occurrences of  $\mathcal{M}$  under a null generative model. This approach has been widely used in the literature since the early 2000s [28, 40], but nonetheless has significant drawbacks. The proper choice of the null model is a concern that was raised soon after the publication of the seminal work of Milo et al. [28], see the comment by Artzy et al. [1]. A suitable null model should generate networks similar to the input graph, as otherwise there is a danger of incorrectly assessing a motif as statistically significant (or not) due to an ill-posed null hypothesis. Also importantly, the current approach suffers from *combinatorial artifacts*. As observed originally by Lior Pachter in his blog [32], as well as by Grochow and Kellis [17], the existence of large independent sets and large cliques can obfuscate the relevance of the count of a motif. Consider the motif  $\mathcal{M}$  with fifteen nodes corresponding to proteins shown in Figure 1(b) on the left as originally shown in [17]. A node connected with a line to a set of nodes enclosed by a circle/oval denotes that the

node is connected to all the nodes within that set. The closed circle/oval shows the topology of the set of nodes within it. For example, we observe that the node in the middle left is connected to three isolated nodes, whereas the two nodes in the middle (both left and right) are connected to four nodes that form a  $K_4$ . Figure 1(b) on the right shows the input network. Due to the existence of a large independent set, and a large clique, the number of occurrences of  $\mathcal{M}$  is equal to  $\binom{12}{3} \times \binom{9}{4}$ . Such a high count may lead to the misleading assessment that  $\mathcal{M}$  is statistically significant. Indeed, combinatorial artifacts occur frequently in real-world networks, which often contain large cliques and independent sets, similar to Figure 1(b).

In this work we contribute towards understanding the motif structure of a network (directed or undirected) in the following ways:

- We propose the novel concept of an  $(f, q)$ -spanning motif. Specifically, a motif is  $(f, q)$ -spanning if there exists a subset of nodes  $S$  that induces an  $f$ -fraction of the motifs, while being a  $q$ -fraction of the node set  $V$ . Intuitively, if  $f$  is close to 1, and  $q$  is close to 0, the motif is likely to be a combinatorial artifact. Based on dense subgraph discovery tools [15], we propose a heuristic algorithm that allows us to test in near-linear time whether a motif is  $(f, q)$ -spanning.
- We propose MOTIFSCOPE, a novel framework that leverages frequently and infrequently appearing motifs to find anomalies in real-world networks. Our framework uses heuristics to find a subgraph that induces many copies of a motif  $\mathcal{M}_2$  and few copies of a motif  $\mathcal{M}_1$ . We show that our framework allows us to find anomalies in social and trust networks.
- We perform an extensive experimental evaluation of various classical and state-of-the-art generative models as null models for assessing statistical significance, which highlights their similarities and differences, as well as the importance of choosing the models.

## 2 Related Work

**Motifs.** A motif is typically a subgraph of constant size. The goal of understanding the motif structure of a network spans numerous disciplines, ranging from systems biology [51] to social network analysis [47] and socio-economics [55], as it sheds light into the building blocks of networks [28]. Motifs have found various algorithmic and machine learning applications, under the umbrella of higher order methods [23, 2, 46, 52].

**Assessing the statistical significance of a motif.** The *de facto* approach for deciding if a motif  $\mathcal{M}$  is statistically significant or not relies on comparing its frequency  $f_{\mathcal{M}}$  to its expected frequency in a null random graph model [28]. While other approaches to assessing the statistical significance of motifs have been proposed, e.g., [4]; in this work we focus on the prevalent approach as introduced by Milo et al. [28]. Given the null model, one samples a large number of networks with the same number of nodes, and counts the frequency of  $\mathcal{M}$ ; let  $\bar{f}_{\mathcal{M}}$ ,  $\sigma_{\mathcal{M}}$  be the average number of occurrences of  $\mathcal{M}$  and the sample standard deviation, respectively. The  $z$ -score is defined as

$$z\text{-score}(\mathcal{M}) = z_{\mathcal{M}} = \frac{f_{\mathcal{M}} - \bar{f}_{\mathcal{M}}}{\sigma_{\mathcal{M}}}.$$

Observe that the  $z$ -score of a motif can be negative; motifs that have a large negative score, and thus appear less often than expected, are sometimes referred in the literature as *anti-motifs* [28, 29].

An important issue is the choice of the null model. A common choice is the configuration model, or one of its variants [5, 14, 10]. This family of models generates a random (di)graph with a given (in-, out-)degree sequence(s). The configuration model was used in the influential works of Milo et al. [28, 29]. However, their approach has received valid critique for a variety of reasons, such as the lack of spatial characteristics [20, 1].

**The densest subgraph problem** aims to find the subgraph with the maximum average degree over all possible subgraphs [16, 8]. Higher-order extensions have been recently proposed that maximize the average density of a small motif such as a triangle [44, 30]. For this problem, as long as the number of nodes in the small subgraph is constant, there exist both efficient polynomial time exact algorithms [44], and faster greedy approximation algorithms [6, 8].

**Graph-based Anomaly Detection** is an intensively active area of graph mining [31], with diverse industrial and scientific applications. We discuss related works in greater detail in the Appendix.

### 3 How to Address Combinatorial Artifacts?

**Problem definition.** As discussed in Figure 1(b), the significance of the motif on the left hand side does not truly represent statistically significant recurring independent motifs, but rather this motif arises because of a combinatorial artifact [32]. It appears around 30 000 times in a PPI network of *S. cerevisiae*, while its occurrences are concentrated into less than 30 nodes. To help clarify such situations, we provide the following definition.

**Definition 1.** A motif  $\mathcal{M}$  is  $(f, q)$ -spanning in graph  $G(V, E)$  if there exists a set of nodes  $S \subseteq V$  such that  $|S| \leq q|V|$  and the induced subgraph  $G[S]$  contains an (at least)  $f$ -fraction of the occurrences of  $\mathcal{M}$  in  $G$ .

We will (loosely) say the statistical significance of a motif  $\mathcal{M}$  according to some null generative model is a *combinatorial artifact* if it is an  $(f, q)$ -spanning motif in  $G(V, E)$  with  $q \ll 1$ , and  $f$  close to 1.<sup>5</sup>

<sup>5</sup> It is worth outlining that forcing  $f = 1$ , and thus simplifying the definition above to a  $(1, q)$ - or just  $q$ -spanning motif is not a robust in the following sense. Consider a graph that is the union of a linear number of node disjoint triangles, and a clique of order  $\sqrt{n}$ . Each node in the graph participates in a triangle, and thus when  $f = 1$ , then  $q = 1$ . However, notice that most of the triangle occurrences appear in the

Our definition of an  $(f, q)$ -spanning motif naturally introduces the following optimization problem.

*Problem 1.* Given a motif  $\mathcal{M}$  and a graph  $G(V, E)$ , what is the largest possible fraction  $f$  of occurrences of  $\mathcal{M}$  among all subgraphs with (at most)  $q|V|$  nodes for a given value of  $q$ ?

We implicitly assume that the motif  $\mathcal{M}$  appears frequently in the graph, and has been assessed statistically significant according to some null generative model; our goal is to understand whether its (apparent) significance is due to a combinatorial artifact or not.

**Hardness.** Problem 1 is NP-hard, and this holds both when we require  $S \subseteq V$  to have exactly  $k = q|V|$  nodes, and at most  $k$  nodes. The reduction is straightforward, and we omit all details. The idea of the proof is that if we could solve Problem 1, then by setting the motif  $\mathcal{M}$  to be a simple undirected edge, we would be able to solve densest- $k$ -subgraph (DkS) problem, and the densest-at-most- $k$ -subgraph (DamkS) problems respectively. Furthermore, we know that these two problems are close in terms of approximation guarantees: if there exists an  $\alpha$ -approximation algorithm for the DamkS problem, then there exists an  $O(\alpha^2)$  approximation algorithm for the DkS problem. The best known approximation factor for the DkS is  $O(n^{-1/4})$  due to Bhaskara et al. [3].

**Theorem 1.** *Problem 1 is NP-hard.*

We also provide a formulation which aims to optimize  $q$  for a given  $f$ , stated as the next problem.

*Problem 2.* Given a motif  $\mathcal{M}$  with  $m(V)$  total occurrences in a graph  $G(V, E)$ , what is the smallest possible size  $q|V|$  of the union of a set of  $f \cdot m(V)$  occurrences for a given value of  $f$ ?

The results of Chlamtač et al. [9] yield the following corollary.

**Corollary 1 (Theorem 1.1 [9]).** *Problem 2 is NP-hard. Furthermore, there exists an  $O(\sqrt{m(V)})$ -approximation algorithm that runs in polynomial time.*

This corollary relates to their results for the minimum  $p$ -union problem (MpU). Consider a hypergraph where each hyperedge corresponds to an occurrence of a motif. Problem 2 can be restated as a minimum  $p$ -union problem (MpU), with  $p = f \cdot m(V)$ . However, their approximation algorithm is not practical for our purposes as it relies on computing maximum flows or solving linear

---

small clique, i.e.,  $O(\sqrt{n})^3 = O(n^{3/2}) \gg O(n)$ . Thus for  $f = O(\frac{n^{3/2}}{n+n^{3/2}}) = 1 - o(1)$ ,  $q$  suddenly becomes  $O(\frac{\sqrt{n}}{n}) = o(1)$ . Similarly, a graph could have multiple distinct smaller combinatorial artifacts, in which case  $f$  might be a constant further from 1 (e.g., 3 small subgraphs with each around 1/3 of the motif copies).

programs, and we are interested in motifs with a large number of occurrences. We therefore propose a more efficient heuristic that works for both problem variants.

---

**Algorithm 1:** COMBART( $G(V, E), \mathcal{M}, f$ )

---

```

1 Initialize  $S_f^* = \emptyset$  ;
2 Count the total number  $m$  of occurrences of  $\mathcal{M}$  in  $G$ ;
3 while  $m(S_f^*)/m < f \wedge m(V) > 0$  do
4    $S \leftarrow \text{GreedyPeeling}(G, \mathcal{M})$ ;
5    $S_f^* \leftarrow S_f^* \cup S$ ;
6    $E \leftarrow E \setminus E[S_f^*]$  ;
7   Update the motif count  $m(V)$ ;
8   Compute  $m(S_f^*)$ ;
9 / *  $E[S_f^*]$  is the set of edges in the induced subgraph  $G[S_f^*]$  * / ;
10  $q \leftarrow |S_f^*|/|V|$  ;
11 return  $q$  ;
```

---

**Proposed Heuristic.** Our heuristic is based on the polynomially time solvable higher-order extension of the densest subgraph problem (DSP) due to Tsourakakis et al. [44, 30]. Our algorithm is shown in pseudocode as Algorithm 1. The algorithm<sup>6</sup> runs as a black-box a greedy peeling algorithm until an  $f$ -fraction of the motif occurrences in the graph have been covered by the subgraph  $S_f^*$ . In each round, the greedy algorithm provides a  $\frac{1}{|V(\mathcal{M})|}$ -approximation to the optimization problem  $\rho^* = \max_{S \subseteq V} \frac{m(S)}{|S|}$ . Here,  $m(S)$  is the number of induced occurrences of motif  $\mathcal{M}$  in  $S$ . Once the algorithm has covered an  $f$ -fraction of  $\mathcal{M}$ -occurrences in  $G$ , we compute  $q$  as  $|S_f^*|/n$  where  $n$  is the number of nodes in  $G$ .

#### 4 MOTIFSCOPE: Anomaly Detection via Motif Contrasting

A reason statistical significance of motifs is considered a worthwhile issue for study is because it gives us important information about graph structure. Indeed, the existence of subgraphs that occur either frequently or infrequently can have interesting algorithmic implications and applications. Here we consider the problem of using motif counts to determine anomalies in a graph structure, such as a social network. Our results utilize the following natural problem.

*Problem 3.* Given a frequent motif  $\mathcal{M}_1$ , and an occurring but infrequent motif  $\mathcal{M}_2$  in a graph  $G$ , find the subset of nodes  $S \subseteq V$  that maximizes the average density difference

$$\max_{S \subseteq V} \frac{m_2(S)}{|S|} - \frac{m_1(S)}{|S|}.$$

<sup>6</sup> While it aims to solve Problem 2, with minor changes it becomes a heuristic for Problem 1.

Intuitively, an induced subgraph  $G[S]$  that contains many induced copies of  $\mathcal{M}_2$ , but few induced copies of  $\mathcal{M}_1$  differs significantly from the global network  $G$  with respect to those two motifs, and therefore possibly in other interesting ways. To solve Problem 3, we use the dense subgraph discovery framework of Tsourakakis et al. [45] with negative weights. We provide an extension of this approach for contrast of motif structures as follows: each node  $v$  is associated with a score  $score(v)$  that is equal to  $m_2(v) - m_1(v)$ . Intuitively, we want to remove nodes that have a large negative score, and keep nodes with a high positive score. The pseudocode is shown in Algorithm 2. Assuming a method MOTIFCOUNT with time complexity  $f(\mathcal{M})$  for motif  $\mathcal{M}$ , our algorithm runs in  $O(n \log n + m + f(\mathcal{M}))$  time in the standard RAM model.

---

**Algorithm 2:** MOTIFSCOPE ( $G, \mathcal{M}_1, \mathcal{M}_2$ )
 

---

```

1  $m_i(v) = \#$  motifs of type  $\mathcal{M}_i$  node  $v$  is contained in ( $i = 1, 2, v \in V(G)$ );
2  $n \leftarrow |V|$ ;
3  $H_n \leftarrow G$ ;
4 for  $i \leftarrow n$  to 2 do
5     Let  $v$  be the vertex of  $G_i$  of minimum score, i.e.,
        $score(v) = m_2(v) - m_1(v)$  (break ties arbitrarily);
6      $H_{i-1} \leftarrow H_i \setminus v$ ;
7     Update counts  $m_1(v), m_2(v)$  for all  $v \in V$ ;
8 return  $H_j$  that achieves maximum average density  $\frac{m_2(S) - m_1(S)}{|S|}$  among  $H_i$ s,
    $i = 1, \dots, n$ ;
```

---

**Implications and applications.** As a specific and important example of the MOTIFSCOPE algorithm, we explain how it can be used to find dense (near)-bipartite subgraphs. In general, the problem of detecting a dense bipartite subgraph in a graph is NP-hard [25]. Finding such subgraphs is important in practice since large bipartite subgraphs in social and trust networks are known to be rare, and frequently correspond to anomalies, such as a collection of manufactured accounts for illicit uses such as money laundering [43, 33]. To attack this problem using MOTIFSCOPE we leverage the fact that a bipartite subgraph does not contain any triangles ( $K_3$ s), which are otherwise common in social networks, but will probably contain several induced cycles of length 4 ( $C_4$ s), which are otherwise rare in social networks [47]. Therefore we set  $\mathcal{M}_1 = K_3$  and  $\mathcal{M}_2 = C_4$ . While our approach is not guaranteed to output a bipartite graph (or even a near-bipartite graph), we show that on real data optimizing for minimizing  $K_3$ s while maximizing  $C_4$ s often yields a bipartite subgraph in practice. As a rule-of-thumb for using MOTIFSCOPE for anomaly detection applications, we propose either using prior knowledge of important subgraphs (such as with the  $K_3$  and  $C_4$  example above), or by choosing  $\mathcal{M}_1$  to be one of the motifs with high  $z$ -score and  $\mathcal{M}_2$  to be one of the motifs with low  $z$ -score.

## 5 Experiments

**Datasets and Code.** Table 1 summarizes the datasets that we use. We use publicly available datasets from a variety of domains, including biological, social, power, and trust networks. The code was written in Python3. We provide both the code and the datasets anonymously at <https://www.dropbox.com/sh/n01h6z1eex5msbh/AAB50qvPX-J1kG5G0hUsYmkHa?dl=0>.

Dataset	$ V $	$ E $	Description	Directed
<i>S. cerevisiae</i> [54]	759	1 593	PPI	×
<i>C. elegans</i> -PPI [54]	2 018	2 930	PPI	×
<i>C. elegans</i> -brain [51]	219	2 416	Connectome	✓
hamsterster [36]	2 426	1 593	Social	×
Eris1176 [36]	1 176	18 552	Power	×
Bitcoin-OTC [22]	5 881	35 592	Trust	✓
Bitcoin-Alpha [21]	3 783	24 186	Trust	✓
LastFM [38]	7 624	27 806	Social	×
Twitch-EN [37]	7 126	35 324	Social	×

Table 1: Summary of datasets.

**Experimental Setup.** The experiments are performed on a single machine, with an Intel i7-10850H CPU @ 2.70GHz and 32GB of main memory. The motif listing algorithm we use is due to Wernicke [50]. We focus on small-sized subgraphs. Figure 2 presents the 13 possible directed motifs of order 3; we shall refer to each motif with their id, for example  $motif_{13}$  is the triangle with all six possible directed edges.

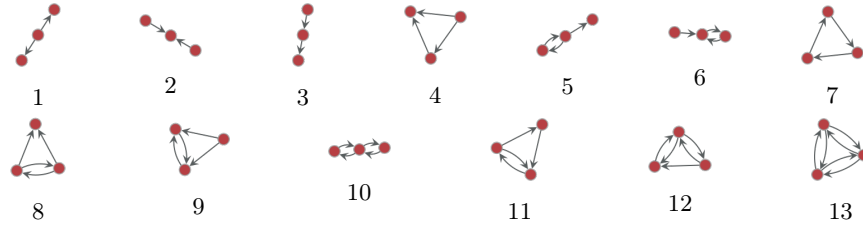


Fig. 2: There exist 13 possible directed motifs of order 3.

### 5.1 Combinatorial artifacts

Table 2 summarizes the performance of COMBART algorithm on five different networks. The second column of the table visualizes a motif of interest  $\mathcal{M}$ . We use a similar notation as [17], where a large node annotated as  $S - c$  ( $K - c$ ) represents an independent set (clique) with  $c$  nodes. We observe that real-world



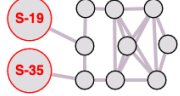
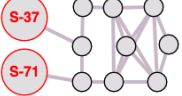


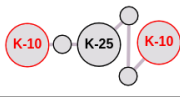

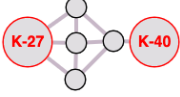
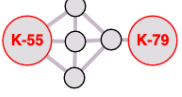

Dataset	Motif	Artifact source	Count	$(f, q)$
<i>S. cerevisiae</i>			$\binom{37}{19} \times \binom{71}{35}$	$(1, 0.06)$
<i>C. Elegans</i> -PPI			$\binom{23}{11} \times \binom{6}{3}$	$(1, 0.063)$
hamsterster			$\binom{21}{10} \times \binom{20}{10}$	$(1, 0.027)$
Eris1176			$\binom{55}{27} \times \binom{79}{40}$	$(1, 0.117)$
<i>C. Elegans</i> -Brain		-	1554	$(0.8, 0.61)$

Table 2: Motifs that are statistically significant from different networks due to combinatorial artifacts. Subgraphs the motifs are clustered in are also listed together with other statistics.

networks typically contain large cliques and independent sets, and thus there exist various motifs whose significance will be a combinatorial artifact. The third column summarizes the subgraph which causes the combinatorial artifact, while the fourth and fifth columns show the motif count which happens to be also the global count ( $f = 1$ ), and the  $(f, q)$  values. As we observe, our novel definition sheds light into assessing the significance of those motifs, by noting that  $f = 1$  and  $q$  is a small fraction of the node set. In contrast, the FFL motif, which is known to play a biological role, is  $(0.8, 0.61)$ -spanning, indicating statistical significance is not due to a combinatorial artifact. We believe these examples show our proposed method can be a significant enhancement to the current approach of assessing the statistical significance of motifs.

## 5.2 MOTIFSCOPE case studies

We show two case studies of MOTIFSCOPE. The first is an algorithmic application that attacks an NP-hard problem using prior knowledge about the appearance of motifs  $\mathcal{M}_1, \mathcal{M}_2$ , while the second application first analyzes the network to choose  $\mathcal{M}_1, \mathcal{M}_2$ .

**Bipartite Subgraphs in Social Networks** As we mentioned in Section 4, we run MOTIFSCOPE using  $\mathcal{M}_1 = K_3, \mathcal{M}_2 = C_4$ , aiming to find a subgraph that induces many cycles of length 4, and few triangles. Our results are summarized in Table 3 for four datasets. We report the total number of induced edges, and

Dataset	# edges	# nodes in $L$	# nodes in $R$
LastFM	124	21	37
Bitcoin-Alpha	24	5	9
Bitcoin-OTC	31	6	10
Twitch-EN	61	7	23

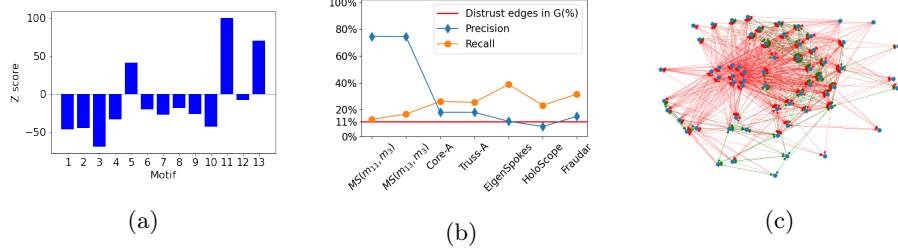
Table 3: Bipartite subgraph found by contrasting  $C_4$  and  $K_3$ .

Fig. 3: Results on the Bitcoin-OTC network. (a) When no prior knowledge is available, we use the  $z$ -scores. Here, we show the  $z$ -scores of the 13 motifs of order 3. (b) Precision and recall for various anomaly detection methods and MOTIFSCOPE (MS) using as  $(\mathcal{M}_1, \mathcal{M}_2)$  motifs  $(motif_{11}, motif_3)$ , and  $(motif_{13}, motif_3)$ , see Figure 2 for the actual motifs. (c) Subgraph found by MOTIFSCOPE for  $(motif_{11}, motif_3)$ . Distrust relations are colored red, and trust relations are colored green.

the number of nodes in the bi-partition  $(L, R)$  of the output node set. Even though our method is not guaranteed to output bipartite subgraphs, the output subgraphs here were in fact all bipartite, i.e., all reported edges having one endpoint in  $L$  and one in  $R$ .

**Anomaly Detection in Trust Networks** We use the Bitcoin-OTC network to illustrate the use of MOTIFSCOPE for anomaly detection on real-world networks. In the Appendix we provide additional results for the Bitcoin-alpha network and camouflage behaviors discovered by MOTIFSCOPE. Since we have no prior knowledge about the motifs in Bitcoin-OTC, we consider all motifs of order 3, and we compute their  $z$ -scores. Figure 3a shows the  $z$ -scores of all 13 motifs. We observe that motif 3 has the most negative  $z$ -score indicating that it appears significantly less often than what we would expect in the directed configuration model. On the contrary, motifs 11, and 13 appear significantly more often. Thus, we use each of motifs 11 and 13 for  $\mathcal{M}_1$ , and motif 3 for  $\mathcal{M}_2$ .

The whole Bitcoin-OTC network contains 11% negative edges, which denote distrust. Figure 3b shows the precision and recall for MOTIFSCOPE, and popular graph anomaly detection methods that use dense subgraph discovery methods, including Core-A and Truss-A from Corescope [41], EigenSpokes [34], Holoscope [26], and Fraudat [18]. Here, we measure the quality of a subgraph  $S$ , using: (i) the precision, namely the fraction of negative edges induced by  $S$  over the total number of edges in  $S$ , and (ii) the recall, namely the fraction of negative

edges in  $S$  over the number of negative edges in the whole graph. We observe that our method outperforms competitors, finding subgraphs that induce a lot of distrust. Figure 3c visualizes one such subgraph. It is worth noting that motifs 11 and 13 are strongly connected, indicating that in this dataset reciprocal edges correlate with trust, whereas motif 3 is a directed chain that lacks reciprocity and correlates with distrust.

**Running times.** Since our graphs are small to medium size, the main computational bottleneck comes from computing motifs on a large ensemble of sampled graphs from the null models. For instance, for Bitcoin-OTC, listing all motifs of order 3 takes around 20 seconds per sampled graph, and the dense subgraph discovery process (greedy peeling [8]) takes around 17 seconds.

## 6 Motif Significance and Null Models

As we have seen, the calculation of statistical significance depends on an underlying null model. In this section we study the following questions, to better understand similarities and differences among frequently used null models.

- Q1 How robust is the significance (or lack thereof) of a given motif  $\mathcal{M}$  across different null models? Is there a consensus between different null models on whether a motif is significant or not?
- Q2 What are the sets of motifs that are statistically significant for different null models, and how do these sets compare to each other? How similar are they with respect to ranking motifs according to their  $z$ -scores?
- Q3 How many samples do we need to generate from a null model, in order to obtain a concentrated estimate of the expected motif count? Is this sample size motif-dependent?

In looking at these questions, We consider seven null models summarized in Table 4 and all 13 motifs of order three in Figure 2. The answer for Q3 is provided in the Appendix due to space constraints. We compare the null models to the well studied *C. elegans* connectome. The network consists of 219 neurons and 2 416 synapses that are represented as nodes and edges respectively, see also Table 1. The network we use corresponds to the adulthood of the *C. elegans*, and was obtained via high-resolution electron microscopy by [51]. All seven generative models we use are well-established in the literature, and they span a period of time from the origins of random graph theory to the most recent advances that involve deep-learning inspired models. Furthermore, we use graph models with independent edge probabilities and dependent edge probabilities. Considering both types of models is important as it was recently shown that random graph models where each edge is added to the graph independently with some probability are inherently limited in their ability to generate graphs with high triangle and other subgraph densities [7]. Furthermore, for any sparse graph, the configuration model is unlikely to generate a large clique. In contrast, it is known that biological networks tend to contain cliques and independent

sets [32]. For this reason, we also use state-of-the-art non-independent models including the prescribed  $k$ -core model (KC) [48], and GraphRNN [53]. For a detailed description of the models, see the Appendix (supplementary material).

Null Models
Directed Erdős-Rényi model (ER) [13]
Edge swap configuration model (ES) [19]
Chung-Lu model (CL) [11]
Partially directed configuration model (PD) [42]
Stochastic Kronecker graphs (KG) [24]
Prescribed $k$ -core model (KC) [48]
GraphRNN (GRNN) [53]

Table 4: Null models used in our experiments, along with their abbreviation. The first five models are *edge independent*, i.e., each edge  $\{i, j\}$  exists independently from the rest with some probability  $p_{ij}$ , while KC and GRNN are not.

**Is there consensus among null models? Mostly no.** We use the *de facto* approach as described in Section 2 to test whether a motif  $\mathcal{M}$  appears more often than expected (i.e.,  $\mathcal{M}$  is a statistically significant motif), or less often than expected (i.e.,  $\mathcal{M}$  is a statistically significant anti-motif) with respect to each of the seven null models. For each null model, we ensure that we have obtained enough samples for a concentrated estimate of the expectation of each motif  $\mathcal{M}$  in Figure 2, by requiring that the coefficient of variation  $CV^2 = \frac{\sigma_{f_{\mathcal{M}}}^2}{f_{\mathcal{M}}^2}$  is at most  $10^{-2}$ ; the weak law of large numbers guarantees concentration, and is a direct application of Chebyshev’s inequality.

For each motif  $\text{motif}_i, i = 1, \dots, 13$  we compute the percentage of the null models that assess it as a statistically significant motif (type A), and anti-motif (type B) respectively. Figure 4(a) summarizes our results. For example, motif 11 is assessed as a type A motif by one model, and similarly as type B by one model. According to the five other models, it is not statistically significant in either sense. Figures 4(b)-(g) provide a detailed overview of the assessment of each model. Perhaps surprisingly, motif 8 is the single motif that is assessed as statistically significant by all seven models. Previous research on other *C. elegans* datasets have identified motif 8 as statistically significant in both the male and hermaphrodite sexes [12]. One can construct motif 8 from motif 4, the feedforward loop (FFL), by introducing one reciprocal connection. Analysis of several species has shown that reciprocal connections are over-represented in connectomes [39]. Interestingly, we do not find feedforward loops [28] being statistically significant by several null models, and this can serve as a criterion for the quality of null models but with caution. The absence of several motor neurons in the analyzed connectomes could in part explain the reduced significance of FFLs. There is a general hierarchy of neurons in *C. elegans* with sensory neurons often connecting to interneurons and interneurons often connecting to motor neurons. Although prior research finds the significance of FFLs within each layer, many of the FFLs did contain one neuron of each type [35].

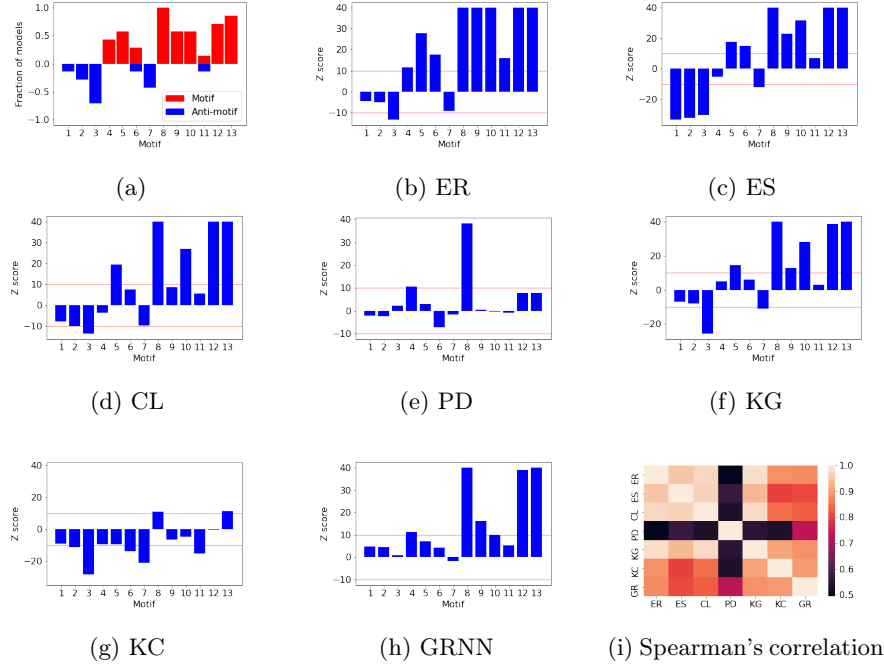


Fig. 4: (a) Histogram of models report each subgraph of size 3 as motif or anti-motif. (b)-(h) Motif significance with respect to  $z$ -score by different random graph models. Plots are clipped at a max value of 40. (i) Pairwise Spearman's correlation coefficient of motif  $z$ -scores of seven models.

**Do null models' rankings agree?** Figure 4(i) shows Spearman's correlation coefficient of the  $z$ -scores respectively for all pairs of null models. The results are illustrated as a heatmap with the similarity scale on the right. We see that the partially directed configuration model is distinctively different from the rest of the 6 models. We explain this difference due to the fact that *C. elegans* has lots of reciprocal directed arcs, i.e., undirected edges, and thus it can model this aspect better than other models in sparse graphs. We observe that variants of the configuration model are not necessarily similar, a point raised by [14]. GraphRNN produces qualitatively similar results to the partially directed configuration model, but the  $z$ -scores are larger due to the fact that the directed version does not capture the frequency of reciprocal edges, despite the wide search of hyperparameters we performed (all details are included in the code).

In a nutshell, caution is required when choosing a null model. Non-independent models, such as the KC and GRNN models, can possibly model complex dependencies that create independent sets and cliques, as described in [7]. GraphRNN seems to be a promising null model for modeling connectomes, although it may not scale well to larger graphs.

## 7 Conclusion

Understanding the importance of motifs in networks is a key problem in connectomics, with a wide range of applications ranging from social network analysis to machine learning. In this work we introduce the novel concept of an  $(f, q)$ -spanning motif that addresses the major issue of *combinatorial artifacts*. We show that determining the smallest value of  $q$  for which there exists a node set of cardinality (at most)  $q|V|$  that induces an  $f$  fraction of the motifs is NP-hard, and we design an efficient heuristic based on dense subgraph discovery methods. Furthermore, we provide new insights into the importance of the null model choice by an extensive empirical analysis of classic and state-of-the-art generative models. Finally, we design the MOTIFSCOPE framework that uses the motif structure of a graph to detect anomalies.

Our work opens several interesting directions. What are the best non-independent edge models as a null model choice? There is an ongoing line of research, with graph RNNs being a recent example [7, 53]. Can we develop new generative models that leverage motifs for *C. Elegans* and model its temporal evolution, see also [47]?

## References

1. Artzy-Randrup, Y., Fleishman, S.J., Ben-Tal, N., Stone, L.: Comment on " network motifs: simple building blocks of complex networks" and " superfamilies of evolved and designed networks". *science* **305**(5687), 1107–1107 (2004)
2. Benson, A.R., Gleich, D.F., Leskovec, J.: Higher-order organization of complex networks. *Science* **353**(6295), 163–166 (2016)
3. Bhaskara, A., Charikar, M., Chlamtac, E., Feige, U., Vijayaraghavan, A.: Detecting high log-densities: an  $o(n^{-1/4})$  approximation for densest k-subgraph. In: Proc. STOC '10. pp. 201–210 (2010)
4. Bloem, P., de Rooij, S.: Large-scale network motif analysis using compression. *Data Mining and Knowledge Discovery* **34**(5), 1421–1453 (2020)
5. Bollobás, B.: A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *European Journal of Combinatorics* **1**(4), 311–316 (1980)
6. Boob, D., Gao, Y., Peng, R., Sawlani, S., Tsourakakis, C., Wang, D., Wang, J.: Flowless: extracting densest subgraphs without flow computations. In: Proc. TheWebConf '20. pp. 573–583 (2020)
7. Chanpuriya, S., Musco, C., Sotiropoulos, K., Tsourakakis, C.: On the power of edge independent graph models. *Advances in NeurIPS* **34** (2021)
8. Charikar, M.: Greedy approximation algorithms for finding dense components in a graph. In: APPROX. pp. 84–95. Springer (2000)
9. Chlamt'ač, E., Dinitz, M., Konrad, C., Kortsarz, G., Rabanca, G.: The densest k-subhypergraph problem. *arXiv preprint arXiv:1605.04284* (2016)
10. Chung, F., Chung, F.R., Graham, F.C., Lu, L., Chung, K.F., et al.: Complex graphs and networks. No. 107, American Mathematical Soc. (2006)
11. Chung, F., Lu, L.: The average distances in random graphs with given expected degrees. *PNAS* **99**(25), 15879–15882 (2002)
12. Cook, S.J., et al.: Whole-animal connectomes of both *caenorhabditis elegans* sexes. *Nature* **571**(7763), 63–71 (2019)

13. Erdős, P., Rényi, A.: On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci* **5**(1), 17–60 (1960)
14. Fosdick, B.K., Larremore, D.B., Nishimura, J., Ugander, J.: Configuring random graph models with fixed degree sequences. *Siam Review* **60**(2), 315–355 (2018)
15. Gionis, A., Tsourakakis, C.E.: Dense subgraph discovery: Kdd 2015 tutorial. In: *Proc. KDD '15*. pp. 2313–2314 (2015)
16. Goldberg, A.V.: Finding a maximum density subgraph. University of California Berkeley, CA (1984)
17. Grochow, J.A., Kellis, M.: Network motif discovery using subgraph enumeration and symmetry-breaking. In: Speed, T., Huang, H. (eds.) *RECOMB*. pp. 92–106 (2007)
18. Hooi, B., Song, H.A., Beutel, A., Shah, N., Shin, K., Faloutsos, C.: Fraudar: Bound-ing graph fraud in the face of camouflage. In: *Proc. KDD '16*. p. 895–904 (2016)
19. Kannan, R., Tetali, P., Vempala, S.: Simple markov-chain algorithms for generating bipartite graphs and tournaments. *Random Struct. Algorithms* **14**(4), 293–308 (1999)
20. King, O.D.: Comment on “subgraphs in random networks”. *Physical Review E* **70**(5), 058101 (2004)
21. Kumar, S., Hooi, B., Makhija, D., Kumar, M., Faloutsos, C., Subrahmanian, V.: Rev2: Fraudulent user prediction in rating platforms. In: *Proc. WSDM '18*. pp. 333–341. ACM (2018)
22. Kumar, S., Spezzano, F., Subrahmanian, V., Faloutsos, C.: Edge weight prediction in weighted signed networks. In: *ICDM*. pp. 221–230. IEEE (2016)
23. Lee, J.B., Rossi, R.A., Kong, X., Kim, S., Koh, E., Rao, A.: Graph convolutional networks with motif-based attention. In: *Proc. CIKM '19*. pp. 499–508 (2019)
24. Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C., Ghahramani, Z.: Kroe-necker graphs: An approach to modeling networks. *J. Mach. Learn. Res (JMLR)* **11**, 985–1042 (2010)
25. Lin, B.: The parameterized complexity of the k-biclique problem. *Journal of the ACM (JACM)* **65**(5), 1–23 (2018)
26. Liu, S., Hooi, B., Faloutsos, C.: Holoscope: Topology-and-spike aware fraud detec-tion. In: *Proc. CIKM '17*. p. 1539–1548 (2017)
27. Mangan, S., Alon, U.: Structure and function of the feed-forward loop network motif. *PNAS* **100**(21), 11980–11985 (2003)
28. Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U.: Network motifs: Simple building blocks of complex networks. *Science* **298**(5594), 824–827 (2002). <https://doi.org/10.1126/science.298.5594.824>
29. Milo, R., Itzkovitz, S., Kashtan, N., Levitt, R., Shen-Orr, S., Ayzenshtat, I., Sheffer, M., Alon, U.: Superfamilies of evolved and designed networks. *Science* **303**(5663), 1538–1542 (2004). <https://doi.org/10.1126/science.1089167>
30. Mitzenmacher, M., Pachocki, J., Peng, R., Tsourakakis, C., Xu, S.C.: Scalable large near-clique detection in large-scale networks via sampling. In: *Proc. KDD '15*. pp. 815–824. ACM (2015)
31. Noble, C.C., Cook, D.J.: Graph-based anomaly detection. In: *Proc. KDD '03*. pp. 631–636 (2003)
32. Pachter, L.: Why i read the network nonsense papers. <https://liorpachter.wordpress.com/2014/02/12/why-i-read-the-network-nonsense-papers/>
33. Pandit, S., Chau, D.H., Wang, S., Faloutsos, C.: Netprobe: a fast and scalable system for fraud detection in online auction networks. In: *WWW* (2007)

34. Prakash, B.A., Sridharan, A., Seshadri, M., Machiraju, S., Faloutsos, C.: Eigen-spokes: Surprising patterns and scalable community chipping in large graphs. In: *Advances in KDD*. pp. 435–448. Springer Berlin Heidelberg (2010)
35. Reigl, M., Alon, U., Chklovskii, D.B.: Search for computational modules in the *c. elegans* brain. *BMC biology* **2**(1), 1–12 (2004)
36. Rossi, R.A., Ahmed, N.K.: The network data repository with interactive graph analytics and visualization. In: *AAAI* (2015), <https://networkrepository.com>
37. Rozemberczki, B., Allen, C., Sarkar, R.: Multi-scale attributed node embedding (2019)
38. Rozemberczki, B., Sarkar, R.: Characteristic Functions on Graphs: Birds of a Feather, from Statistical Descriptors to Parametric Models. In: *Proc. CIKM '20*. p. 1325–1334 (2020)
39. Scheffer, L.K., et al.: A connectome analysis of the adult drosophila central brain. *Elife* **9** (2020)
40. Shen-Orr, S., Milo, R., Mangan, S., Alon, U.: Network motifs in the transcriptional regulation network of *escherichia coli*. *Nature genetics* **31**, 64–8 (06 2002)
41. Shin, K., Eliassi-Rad, T., Faloutsos, C.: Corescope: graph mining using k-core analysis: patterns, anomalies and algorithms. In: *ICDM '16*. pp. 469–478 (2016)
42. Spricer, K., Britton, T.: The configuration model for partially directed graphs. *Journal of Statistical Physics* **161**, 965–985 (2015)
43. Starnini, M., et al.: Smurf-based anti-money laundering in time-evolving transaction networks. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. pp. 171–186. Springer (2021)
44. Tsourakakis, C.: The k-clique densest subgraph problem. In: *Proc. WWW '15*. pp. 1122–1132 (2015)
45. Tsourakakis, C.E., Chen, T., Kakimura, N., Pachocki, J.: Novel dense subgraph discovery primitives: Risk aversion and exclusion queries. In: *Proc. ECML PKDD '19*. pp. 378–394. Springer (2019)
46. Tsourakakis, C.E., Pachocki, J., Mitzenmacher, M.: Scalable motif-aware graph clustering. In: *Proc. WWW '17*. pp. 1451–1460 (2017)
47. Ugander, J., Backstrom, L., Kleinberg, J.: Subgraph frequencies: Mapping the empirical and extremal geography of large graph collections. In: *Proc. WWW '13*. pp. 1307–1318 (2013)
48. Van Koeveing, K., Benson, A., Kleinberg, J.: Random graphs with prescribed k-core sequences: A new null model for network analysis. In: *Proc. TheWebConf '21*. p. 367–378 (2021)
49. Wasserman, S., Faust, K., et al.: *Social network analysis: Methods and applications* (1994)
50. Wernicke, S., Rasche, F.: Fanmod: a tool for fast network motif detection. *Bioinformatics* **22**(9), 1152–1153 (2006)
51. Witvliet, D.e.a.: Connectomes across development reveal principles of brain maturation. *Nature* **596**(7871), 257–261 (2021)
52. Yin, H., Benson, A.R., Leskovec, J., Gleich, D.F.: Local higher-order graph clustering. In: *Proc. KDD '17*. pp. 555–564 (2017)
53. You, J., Ying, R., Ren, X., Hamilton, W.L., Leskovec, J.: Graphrnn: Generating realistic graphs with deep auto-regressive models. In: *ICML* (2018)
54. Yu, H., et al.: High-quality binary protein interaction map of the yeast interactome network. *Science (New York, N.Y.)* **322**, 104–10 (09 2008)
55. Zhang, X., Shao, S., Stanley, H., Havlin, S.: Dynamic motifs in socio-economic networks. *EPL (Europhysics Letters)* **108** (12 2014)