The Common-Neighbors Metric Is Noise-Robust and Reveals Substructures of Real-World Networks

Sarel Cohen[1], Philipp Fischbeck[2(✉)], Tobias Friedrich[2], and Martin Krejca[3]

[1] The Academic College of Tel Aviv-Yaffo, Tel Aviv, Israel
sarelco@mta.ac.il
[2] Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
{philipp.fischbeck,tobias.friedrich}@hpi.de
[3] LIX, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris, Palaiseau, France
martin.krejca@polytechnique.edu

Abstract. Real-world networks typically display a complex structure that is hard to explain by a single model. A common approach is to partition the edges of the network into disjoint simpler structures. An important property in this context is *locality*—incident vertices usually have many common neighbors. This allows to classify edges into two groups, based on the number of the common neighbors of their incident vertices. Formally, this is captured by the *common-neighbors* (CN) metric, which forms the basis of many metrics for detecting *outlier* edges. Such outliers can be interpreted as noise or as a substructure.

We aim to understand how useful the metric is, and empirically analyze several scenarios. We randomly insert outlier edges into real-world and generated graphs with high locality, and measure the metric accuracy for partitioning the combined edges. In addition, we use the metric to decompose real-world networks, and measure properties of the partitions. Our results show that the CN metric is a very good classifier that can reliably detect noise up to extreme levels (83% noisy edges). We also provide mathematically rigorous analyses on special random-graph models. Last, we find the CN metric consistently decomposes real-world networks into two graphs with very different structures.

Keywords: Noise · Clustering · Networks

1 Introduction

The structure of real-world processes across a large variety of scientific domains, such as biology, ecology, sociology, or technology, typically results in highly complex networks [3,14]. These networks display many structural properties, such as high *heterogeneity* (many different vertex degrees) and high *locality* (vertices that

H. Kashima et al. (Eds.): PAKDD 2023, LNAI 13935, pp. 67–79, 2023.
https://doi.org/10.1007/978-3-031-33374-3_6

share a large common neighborhood are likely to be connected), which seem to play a crucial role for reasoning about the networks [4]. Thus, it comes as no surprise that these properties are utilized in order to decompose complex networks into simpler ones. A prominent approach for this task is *graph clustering* [18].

Graph clustering aims to partition the vertices of a network into sets such that vertices from the same set have a similar value based on some metric, for example, the nearest neighbors of each vertex [22]. An important special case of clustering, also typically performed as a pre-processing step in clustering [6], is *outlier detection* [1], which aims to separate vertices with suspicious metric values from the rest. Algorithms for outlier detection vary in the amount of information they utilize. Some settings consider graphs annotated with features [10,13,15]. Other settings work exclusively with the structure of the network, that is, its vertices and edges [20]. Many approaches define a metric for *vertices* [9].

An alternative approach is to classify the *edges* of a network instead of its vertices [2]. In this setting, outlier detection is the opposite of link prediction [12]. Results for edge outlier detection are scarce, with the article by Zhang, Kiranyaz, and Gabbouj [21] being the most extensive one. The authors consider different edge metrics based on the *common-neighbors* (CN) metric, which counts the number of shared vertices of the two vertices incident to a given edge. The authenticity of an edge is determined by how largely its metric score differs from the expected score of an edge, assuming the outlier-free graph follows a certain random-graph distribution. This approach is evaluated on real-world networks with randomly added edges. Although the real-world networks do not necessarily match the theoretical assumptions required for the authenticity of an edge, the authors show that their different metrics typically achieve an area-under-ROC-curve value of at least 0.85. This shows these metrics are rather robust to noise, making edge outlier detection a promising tool for *noise detection* in networks.

Contribution. Motivated by the good performance of the metrics by Zhang, Kiranyaz, and Gabbouj [21], we focus on the usefulness of the *pure* CN metric for edge outlier detection, that is, we use the CN metric without *any* assumptions about the underlying graph model. Our intention is to use the CN metric in order to partition the edge set of a graph into two sets, each of which represents the connections of a different graph. Ideally, the two resulting graphs differ in locality, a very defining graph property, as we remarked above. If one of the resulting graphs is close to a (random) noise graph, then our setting resembles noise detection in graphs. However, it is more general than that, as we do not require any of the graphs to follow a noise model.

Setting. We consider *mixed graphs*, which are the superposition of two graphs defined over the same set of vertices but with different edge sets. One of the two graphs that make up the mixed graph is the *base graph*, which we consider to be the graph that consists of no outlier edges. The other graph is the *overlay graph*. We apply the CN metric to the mixed graph and evaluate how well it can separate the base graph from the overlay graph.

Methodology. We evaluate the performance of the CN metric empirically in three different settings (Sect. 4). In the first setting (Sect. 4.1), both the base and the overlay graph follow well-established random-graph distributions. As base graphs, we use graphs that place their vertices randomly with respect to a geometry: random geometric graphs [16] and hyperbolic random graphs [11]. These models have a high locality, the second one also high heterogeneity. For the overlay graph, we use the Erdős–Rényi model [8], that is, we add edges independently, each with the same probability. As this model does not make use of locality, the separation should work well.

In the second setting (Sect. 4.2), we exchange the base graph for real-world networks. The overlay graph still follows the Erdős–Rényi model. Thus, the edges of the overlay graph remain to not follow any locality. Here, we aim to see how sufficient the natural locality of real-world networks is for a good separation.

In the final setting (Sect. 4.3), the *mixed* graph is a real-world network, i.e., we have no ground truth information anymore. The aim is to see how well the CN metric separates a real-world network into two distinct graphs. To this end, we vary the threshold that determines when an edge is classified as an outlier, and we compare graph properties in the resulting base and overlay graphs.

Results. For all three settings, the CN metric performs very well. For the first setting (Sect. 4.1), the CN metric achieves an area-under-ROC-curve (AUC) value of at least 0.96—in many cases of at least 0.98. These results hold even for extreme scenarios where the amount of random/outlier edges is 5 times the amount of edges in the base graph. This shows that the CN metric is immensely robust with respect to non-local noise.

For the second setting (Sect. 4.2), the quality depends more on the base graph, with some settings having a (still rather high) AUC value of 0.80, whereas others have a value of over 0.90. This shows the locality of real-world networks is high enough such that non-local noise is well detected. However, our experiments indicate the quality also depends on other graph properties like graph density.

For the final setting (Sect. 4.3), the number of components as well as the global clustering coefficient (GCC) of the two resulting graphs indicate that the CN metric does indeed classify non-local edges as overlay edges, as the GCC of the base graph increases with the removal of overlay edges, and the number of connected components also quickly increases.

In addition to our empirical results, we prove mathematically rigorously what the expected CN score of an edge in mixed graphs is, with respect to whether the edge was present in the base graph (Theorem 1) or only in the overlay graph (Theorem 2). In these analyses, we assume that the base graph is a random geometric graph and the overlay graph an Erdős–Rényi graph, which we assume to be sparse. We find that the expected difference between the CN score of an edge in the mixed graph that is already present in the base graph versus the score if the edge is only present in the overlay graph is in the order of magnitude of the expected vertex degree of the base graph. Thus, a higher expected vertex degree of the base graph makes it easier to detect outliers.

Conclusion. Our results indicate that the CN metric is very well suited for classifying real-world networks into two distinct, simpler networks. Neither the CN metric nor the classification method require any problem-specific knowledge. Especially, the CN metric is highly robust to noise. This all suggests that the simple CN metric is a very good tool for handling the detection of outlier edges.

2 Preliminaries

Let $\mathbf{N}$ denote the set of all natural numbers (incl. 0). For all $m, n \in \mathbf{N}$, let $[m..n] \coloneqq [m, n] \cap \mathbf{N}$, and let $[n] \coloneqq [1..n]$. We consider undirected, simple graphs $G = (V, E)$, with *vertices* in V and *edges* in E. For all $v \in V$, we denote the *(exclusive) neighborhood of* v by $\Gamma_G(v) = \{u \in V \mid \{u, v\} \in E\}$. Further, let $\binom{V}{2} \coloneqq \{\{u, v\} \mid u, v \in V \wedge u \neq v\}$ denote the set of all unordered pairs over V.

2.1 Setting

We consider *mixed graphs* $G = (V, E)$ that are the superposition a *base graph* $G_\mathrm{b} = (V, E_\mathrm{b})$ and an *overlay graph* $G_\mathrm{o} = (V, E_\mathrm{o})$, that is, $E = E_\mathrm{b} \cup E_\mathrm{o}$. We say that G *is composed of* G_b *and* G_o.

We consider the *common-neighbors (CN) metric*. For a graph $G = (V, E)$, the CN metric (over G) is the function $\mathrm{cn}_G \colon \binom{V}{2} \to [0..|V| - 2]$ that maps each pair of vertices to the size of their shared neighborhood. That is, for all $\{u, v\} \in \binom{V}{2}$, it holds that $\mathrm{cn}_G(\{u, v\}) = |\Gamma_G(u) \cap \Gamma_G(v)|$. Note that u and v are not accounted for, as $u \notin \Gamma_G(u)$ and $v \notin \Gamma_G(v)$. We call $\mathrm{cn}_G(\{u, v\})$ the *CN score* of $\{u, v\}$.

2.2 Random-Graph Models

We consider various formal random-graph models, which we introduce in the following. In addition to those, we also consider (deterministic) real-world networks, which we explain in Sect. 4.2. For all of the following models, when we introduce a graph, it actually represents a random element following a distribution over the set of all graphs that can be constructed as described. This distribution is defined implicitly via the random choices for how the vertices and/or edges are drawn. We do not introduce special notation for such a distribution.

Random Geometric Graphs. A *random geometric graph* (RGG) is a graph $G = (V, E)$ with $V \subset [0, 1]^2$ together with a *radius* $r \in [0, 1/\sqrt{2}]$. The vertices of an RGG lie in the unit torus, that is, for all $u, v \in V$, the distance between u and v is wrapping around the borders, formally, $\mathrm{dist}(u, v) \coloneqq \sqrt{|u_1 - v_1|_\mathrm{o}^2 + |u_2 - v_2|_\mathrm{o}^2}$, where, for all $i \in [2]$, it holds that $|u_i - v_i|_\mathrm{o} \coloneqq \min\{|u_i - v_i|, 1 - |u_i - v_i|\}$.

The vertices of an RGG are placed independently and uniformly at random into the unit torus, that is, the probability for a vertex to be placed in an area of size $A \in [0, 1]$ is A. After placing the vertices, the edges are determined deterministically by connecting two vertices if and only if their distance is at most r. That is, $E = \{\{u, v\} \in \binom{V}{2} \mid \mathrm{dist}(u, v) \leq r\}$. Since a vertex u is

connected to another vertex v if and only if v is in a circle of radius r around u, the expected degree of u is $(|V|-1)\pi r^2$.

Erdős–Rényi Graphs. An *Erdős–Rényi graph* (ER) is a graph $G = (V, E)$ together with an *edge probability* $p \in [0, 1]$. In contrast to an RGG, the vertices of an ER have no geometric interpretation and can be anything. The edges of G are all drawn independently, each with probability p. That is, for each $\{u, v\} \in \binom{V}{2}$, it holds that $\Pr[\{u, v\} \in E] = p$. Since a vertex u is connected to another vertex v with probability p, the expected degree of u is $(|V|-1)p$.

Hyperbolic Random Graphs. A *hyperbolic random graph* (HRG) is a graph $G = (V, E)$ together with a power-law exponent $\beta \in (2, 3)$ and a radius R. All vertices are positioned in a disk of radius R in the hyperbolic plane according to a probability distribution based on β, and two vertices are connected by an edge if and only if their hyperbolic distance is at most R. The expected average degree can be controlled via R, while β determines the exponent of the power-law degree distribution. The resulting graphs have high heterogeneity and locality.

Randomness in Mixed Graphs. When we consider mixed graphs G composed of a base graph G_b and an overlay graph G_o, we make sure that at most one model determines how vertices are placed. This guarantees that no random choices conflict with each other, so G is well-defined. Since G_b and G_o have their own edges, the randomness in drawing the edges cannot conflict with each other.

3 Theoretical Results

We consider mixed graphs $G = (V, E)$ composed of an RGG $G_\mathrm{rgg} = (V, E_\mathrm{rgg})$ with radius $r \in [0, 1/4]$ as base graph and an ER $G_\mathrm{er} = (V, E_\mathrm{er})$ with edge probability $p \in [0, 1]$ as overlay graph. We mathematically analyze the CN score of an edge $e \in E$, depending on whether e is present in the base graph or not (Sect. 3.2). Our main results are Theorems 1 and 2, which show together that for $p = \mathrm{o}(1)$ (with respect to $|V|$), that is, the overlay graph is not dense, the expected difference of the CN score of e with respect to whether it is present in the base graph or not is in the order of nr^2, which is the same order as the expected vertex degree in an RGG. Thus, the higher the expected vertex degree of the base graph, the further the CN scores in the mixed graph differ from edges present in the base graph and those only present in the overlay graph.

Before we introduce and discuss the results, we discuss important properties relevant to the results. These revolve around the probabilities for vertices to lie at a certain distance with respect to two given vertices u and v, whose CN score we are interested in. We omit proofs due to space restrictions.

3.1 Probabilities of Vertex Placements

Let u and v be vertices from a mixed graph $G = (V, E)$ based on an RGG $G_\mathrm{rgg} = (V, E_\mathrm{rgg})$ of radius $r \in [0, 1/4]$ and an ER $G_\mathrm{er} = (V, E_\mathrm{er})$ with edge probability $p \in [0, 1]$. In order to determine how much the CN score of u and v

changes from G_{rgg} to G, we calculate how likely it is for other vertices to have edges to u and v, both in E_{rgg} and in E_{er}. In the following, we first determine the probability of a vertex being connected to both u and v in G_{rgg}. Then, we determine the probability of a vertex that is not a common neighbor of u and v in G_{rgg} to be a common neighbor in G.

Common Neighbors in the Base Graph. In this setting, the shared area of the two circles of radius r around u and v is important. We call this area $\mu(u \cap v)$, and we remark it is the probability of a vertex to be in $\Gamma_{G_{\text{rgg}}}(u) \cap \Gamma_{G_{\text{rgg}}}(v)$, as they are drawn uniformly at random. Based on this, we derive the expectation of $\mu(u \cap v)$ with respect to whether u and v are themselves connected.

Lemma 1. *Let $G_{\text{rgg}} = (V, E_{\text{rgg}})$ be an RGG with radius $r \in [0, 1/4]$. Furthermore, let $\{u, v\} \in \binom{V}{2}$ and let $R \coloneqq \{\{u, v\} \in E\}$. Then*

$$\mathrm{E}[\mu(u \cap v) \mid R] = \frac{4\pi - 3\sqrt{3}}{4} r^2 \text{ and } \mathrm{E}[\mu(u \cap v) \mid \overline{R}] = \frac{3\sqrt{3}\pi r^2}{4(1 - \pi r^2)} r^2. \quad (1)$$

Common Neighbors in the Mixed Graph. We consider the probability of a vertex w to be a common neighbor of u and v in G, given that it is not a common neighbor in G_{rgg}. This happens because of one of the following reasons.

1. $w \in \Gamma_{G_{\text{rgg}}}(u)$: In this case, $w \notin \Gamma_{G_{\text{rgg}}}(v)$. Since $w \in \Gamma_G(u) \cap \Gamma_G(v)$, there is an edge in $E_{\text{er}} \setminus E_{\text{rgg}}$.
2. $w \in \Gamma_{G_{\text{rgg}}}(v)$: This case is symmetric to the previous one when exchanging u with v, as all vertices are handled symmetrically in RGGs.
3. $w \in \overline{\Gamma_{G_{\text{rgg}}}(u) \cup \Gamma_{G_{\text{rgg}}}(v)}$: In this case, there are two edges in $E_{\text{er}} \setminus E_{\text{rgg}}$.

The following lemma determines the probability of w falling into one of these three cases.

Lemma 2. *Let $G = (V, E)$ be a mixed graph composed of an RGG $G_{\text{rgg}} = (V, E_{\text{rgg}})$ with radius $r \in [0, 1/4]$ as base graph and an ER $G = (V, E_{\text{er}})$ with edge probability $p \in [0, 1]$ as overlay graph. Furthermore, let $\{u, v\} \in \binom{V}{2}$ and $w \in V \setminus \{u, v\}$. Last, let O denote the event $\{w \notin \Gamma_{G_{\text{rgg}}}(u) \cap \Gamma_{G_{\text{rgg}}}(v)\}$, and let R denote the event $\{\{u, v\} \in E_{\text{rgg}}\}$. Then, abbreviating $a \coloneqq (3\sqrt{3})/4$,*

$$\Pr[w \in \Gamma_G(u) \cap \Gamma_G(v) \wedge O \mid R] = pr^2\big(2a - (\pi + a)p\big) + p^2 \text{ and} \quad (2)$$

$$\Pr[w \in \Gamma_G(u) \cap \Gamma_G(v) \wedge O \mid \overline{R}] = p\pi r^2 \left(2(1-p) - (2-p)\frac{ar^2}{1 - \pi r^2}\right) + p^2. \quad (3)$$

3.2 The CN Score of Different Edges

Using the probabilities from Sect. 3.1, we derive the expected CN score of an edge in the mixed graph. The following theorem assumes that the edge is already present in the base graph. Afterward, we consider the case that the edge is only present in the overlay graph. At the end, we conclude.

Theorem 1. *Let $G = (V, E)$ be a mixed graph over $n \in \mathbf{N}_{\geq 2}$ vertices composed of an RGG $G_{\text{rgg}} = (V, E_{\text{rgg}})$ with radius $r \in [0, 1/4]$ as base graph and an ER $G = (V, E_{\text{er}})$ with edge probability $p \in [0, 1]$ as overlay graph. Furthermore, let $\{u, v\} \in E$, let $N_G = \text{cn}_G(\{u, v\})$, let $N_{G_{\text{rgg}}} = \text{cn}_{G_{\text{rgg}}}(\{u, v\})$, let q denote the left expected value from Eq. (1), let s denote the probability from Eq. (2), and let R denote the event $\{\{u, v\} \in E_{\text{rgg}}\}$. Then*

$$\mathrm{E}[N_G \mid R] = \mathrm{E}[N_{G_{\text{rgg}}} \mid R] + (n-2)s \text{ and } \mathrm{E}[N_{G_{\text{rgg}}} \mid R] = (n-2)q.$$

The following theorem shows how the CN score changes if the edge is only in the overlay graph. It looks similar to Theorem 1 but considers other probabilities.

Theorem 2. *Let $G = (V, E)$ be a mixed graph over $n \in \mathbf{N}_{\geq 2}$ vertices composed of an RGG $G_{\text{rgg}} = (V, E_{\text{rgg}})$ with radius $r \in [0, 1/4]$ as base graph and an ER $G = (V, E_{\text{er}})$ with edge probability $p \in [0, 1]$ as overlay graph. Further, let $\{u, v\} \in E$, let $N_G = \text{cn}_G(\{u, v\})$, let $N_{G_{\text{rgg}}} = \text{cn}_{G_{\text{rgg}}}(\{u, v\})$, let q denote the right expected value from Eq. (1), let s be the probability from Eq. (3), let R denote the event $\{\{u, v\} \notin E_{\text{rgg}}\}$, and let K denote the event $\{\{u, v\} \in E_{\text{er}}\}$. Then*

$$\mathrm{E}[N_G \mid \overline{R}, K] = \mathrm{E}[N_{G_{\text{rgg}}} \mid \overline{R}, K] + (n-2)s \text{ and } \mathrm{E}[N_{G_{\text{rgg}}} \mid \overline{R}, K] = (n-2)q.$$

Let q_{rgg} and s_{rgg}, respectively, denote q and s from Theorem 1, and let q_{er} and s_{er} be defined analogously with respect to Theorem 2. If $(q_{\text{rgg}} + s_{\text{rgg}})$ and $(q_{\text{er}} + s_{\text{er}})$ are sufficiently separated, then so are the respective CN scores for edges in the mixed graph that are present in the base graph or only in the overlay graph, which makes separating these two edge types not difficult. By Lemma 1, it holds that $q_{\text{rgg}} - q_{\text{er}} = \big(\pi - 3\sqrt{3}/\big(4(1 - \pi r^2)\big)\big)r^2 = \Theta(r^2)$, which is non-negative for all $r \in [0, 1/4]$. Similarly, by Lemma 2, we get that $s_{\text{rgg}} - s_{\text{er}} = -(2-p)pr^2(\pi - (3\sqrt{3})/4 - \pi^2 r^2)/(1 - \pi r^2) = -\Theta(pr^2)$, which is non-positive for all $r \in [0, 1/4]$ and all $p \in [0, 1]$. Due to the difference of the signs, a general comparison is difficult. However, assuming that the overlay graph is sparse, that is, $p = \mathrm{o}(1)$, we see that $(q_{\text{rgg}} + s_{\text{rgg}}) - (q_{\text{er}} + s_{\text{er}}) = \Theta\big(r^2\big)$. Thus, the difference in the expected CN score of edges present in the base graph and those only present in the overlay graph is $\Theta\big(r^2 n\big)$, which is in the same order as the expected vertex degree of an RGG. Thus, an increased average degree in the base graph results in a larger expected difference in scores.

4 Empirical Results

We present empirical findings on the quality of the CN metric for different scenarios. We first consider scenarios where we know both the base graph and the overlay graph. As base graph, we consider two random graph models (Sect. 4.1) as well as real-world networks (Sect. 4.2). Last, we consider the case where the mixed graph is a real-world network, and we partition its edges according to the CN metric (Sect. 4.3). We briefly explain how we carry out our study.

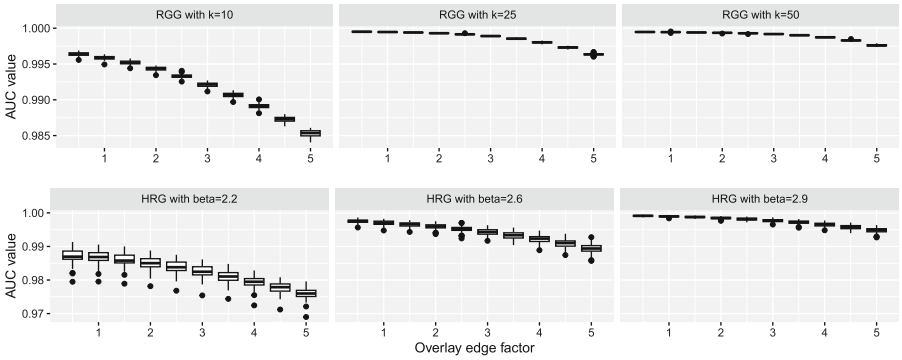

Fig. 1. (Top) The AUC score for an RGG as base graph and an ER as overlay graph. We fix the number of vertices to 5 000 and the expected average degree of the base graph $k \in \{10, 25, 50\}$. The overlay edge factor varies from 0.5 to 5, and we display 50 samples per configuration. (Bottom) The AUC score for an HRG as base graph and an ER as overlay graph. We fix the number of vertices to 5 000, expected average degree of the base graph $k = 25$, and vary the power-law degree exponent $\beta \in \{2.2, 2.6, 2.9\}$. The overlay edge factor varies from 0.5 to 5, and we display 50 samples per configuration.

AUC Metric. When evaluating the quality of the CN metric for separation of the two known edge sets, we measure the well-established *area-under-the-ROC-curve (AUC)* score. This measure is commonly used for classification models and provides an aggregate measure for the true-positive and false-positive rate of a binary classifier across all possible thresholds. We treat our scenario as a binary classification task, with base edges being positive. The AUC essentially is the probability that a random positive example has a higher score than a random negative example, i.e., that the CN score of a random base edge is higher than that of a random overlay edge. A random metric would yield an AUC score of 0.5, while a perfect metric would yield 1.0.

Experimental Setup. Our Python implementation uses the libraries NetworKit [19] and igraph [7] for generating and analyzing graphs. They provide implementations for random graph models and graph properties. All experiments were run on a system with an Apple M1 chip and 16 GB RAM. However, note we do not consider run times, and all experiments were finished in minutes. All code and data is published at https://github.com/PFischbeck/cn-noise-experiments.

4.1 Graph Model as Base Graph

We consider two graph models as base graph, which are known to be highly clustered due to the use of an underlying geometry in the generation process. As base graph, we consider RGGs as well as HRGs (see Sect. 2.2 for details). As overlay graph, we consider ERs with an expected number of edges relative to the number of edges in the base graph. For example, an *overlay edge factor*

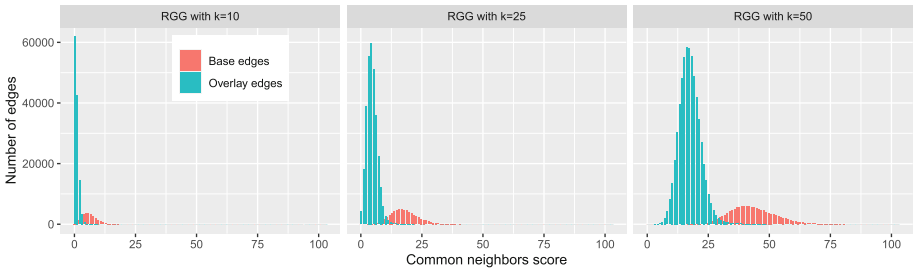

Fig. 2. The distribution of the CN scores for an RGG with 5 000 vertices and varying expected degree k as base graph, and an ER as overlay graph with overlay edge factor 5. The color shows whether the edges are from the base (red) or overlay graph (blue). (Color figure online)

of 2 means that there are twice as many overlay edges as there are base edges, in expectation. For a fixed model configuration and overlay edge factor, we take 50 samples and display them as box plots.

Random Geometric Graphs. For RGGs as base graph, we fix the number of vertices to 5 000 and vary the expected average degree to be 10, 25, and 50. Figure 1 (top) shows the resulting AUC scores for varying overlay edge factors.

One clearly sees that in all scenarios, the AUC score is very high, staying above 0.98. As one would expect, an increased overlay edge factor leads to lower scores, as the overlay edges make it harder to tell the two edge sets apart. The dependence on the average degree seems to consist of two parts. First, there is an increase of the AUC score for increased average degree, as predicted in Sect. 3. In addition, for higher average degree, the increase in overlay edges has a reduced effect on the AUC score. Recall that the number of overlay edges is relative to the number of base edges and thus also scales with increased average degree.

In order to understand this behavior better, we also provide a view on the distribution of scores for the two edge partitions. We fix an overlay edge factor of 5 and look at one sample for all three considered average degrees. Figure 2 shows the score distribution for these configurations.

As the average degree is increased, the CN scores increase for both base and overlay edges. However, they also increase their variance, and thus their overlap increases. Nonetheless, the high average degree still makes it easy to distinguish between the high number of edges outside of the overlap for $k = 50$.

Hyperbolic Random Graphs. For HRGs as base graph, we fix the number of vertices to 5 000, expected average degree $k = 25$, and vary the power-law degree exponent to be 2.2, 2.6, and 2.9. Figure 1 (bottom) shows the resulting AUC scores for varying overlay edge factors.

Across all three configurations, the AUC score is relatively high, although not as high as for the RGGs as base graph (Fig. 1 (top)). Recall that a lower power-law exponent corresponds to a more heterogeneous degree distribution, leading to many low-degree and few high-degree vertices. For base graphs with

Table 1. The real-world networks we use as base graph, with their number of vertices n, their number of edges m, and global clustering coefficient (GCC).

Graph	n	m	GCC
advogato	6 k	43 k	0.11
bio-WormNet-v3-benchmark	2 k	79 k	0.72
ca-HepPh	11 k	118 k	0.66
ia-digg-reply	30 k	86 k	0.02
soc-brightkite	57 k	213 k	0.11
web-indochina-2004	11 k	48 k	0.57

low power-law exponent, edges connected to low-degree vertices have low CN scores, making them harder to differentiate from the overlay edges. This leads to a lower AUC score. Further, a higher overlay edge factor yields a lower AUC score. This is because the CN score of overlay edges is increased by other overlay edges. As the power-law exponent increases, the variance of the AUC score decreases.

4.2 Real-World Network as Base Graph

We consider various real-world networks as base graph, with an ER as overlay graph. The real-world networks are shown in Table 1. They are part of the NetworkRepository collection [17], and we use them in a cleaned format [5]. The networks are from different contexts (including biological, social, and web networks) and vary both in graph size and in their locality. We measure locality via the *global clustering coefficient (GCC)*, which can be interpreted as the probability that a triplet of vertices with at least two edges also has the third edge. Thus, it is an indicator for how clustered or local a graph is.

For every real-world network, we add an ER overlay graph with the same number of vertices as the base graph, and we vary the overlay edge factor from 0.5 to 5. We take 50 samples per configuration (recall that the ER overlay graph is random), and we consider the resulting AUC score of the CN scores.

The AUC scores for almost all real-world networks are at a high level, even with 5 times as many overlay edges as base edges. In addition, for most graphs, the AUC score remains constant as the overlay edge factor varies. The exceptions are the graphs `bio-WormNet-v3-benchmark` and `ca-HepPh`. Both graphs have few vertices and high clustering, which might lead to higher CN scores for overlay edges, both via other overlay edges and base edges.

Overall, there is a strong relation between the graph clustering (via the global clustering coefficient) and the AUC score of the CN metric. The `ia-digg-reply` network has a very low GCC and low AUC scores. Based on our experiments, we think the metric quality depends on several graph properties, including clustering, degree heterogeneity, graph density, and number of low-degree vertices.

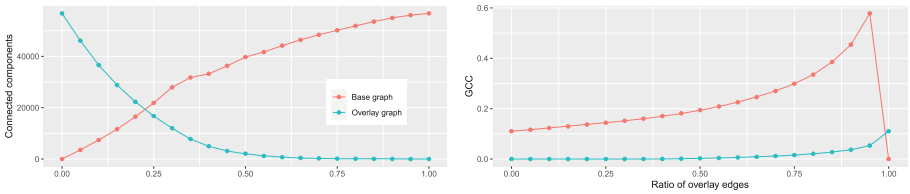

Fig. 3. The number of connected components and the global clustering coefficient of the base graph (red) and overlay graph (blue) when splitting the edges of the `soc-brightkite` network according to the CN metric. The ratio r_{split} defines that the $r_{\text{split}} \cdot |E|$ edges with lowest CN score are classified as overlay edges. (Color figure online)

The results are under the assumption that the real-world base graphs do not contain any overlay edges themselves, which cannot be known. In order to better understand this real-world edge set, we also consider real-world networks as the mixed graph in the following section.

4.3 Real-World Graph as Mixed Graph

In the experiments above, we had control over the base and overlay graph and thus were able to evaluate the quality of the CN score based on this ground truth. However, when partitioning a given graph without ground truth, we have to turn to other properties. In particular, if this metric does indeed help partition the given graph into a local, clustered structure and a global, random structure, this should be reflected in the properties of the two partition sets. We investigate this here. To this end, we take the real-world network `soc-brightkite` and treat it as a mixed graph. We measure the CN scores of its edges and sort the edges according to this score, with ties solved uniformly at random. For a fixed ratio r_{split}, the $r_{\text{split}} \cdot |E|$ edges with the lowest score are classified as overlay edges, while the remaining edges are classified as base edges. We build the base graph and overlay graph according to this edge partitioning, and we measure the global clustering coefficient as well as the number of connected components of the two parts. Figure 3 shows the resulting values for varying ratio r_{split}.

As the split ratio increases, the number of components of the base graph quickly rises, with an average of roughly two vertices per component for $r_{\text{split}} = 0.3$. On the other hand, the number of components of the overlay graph quickly decreases, which indicates that the edges classified as overlay edges are in fact global in the sense that they often connect previously disconnected components.

Also, as more edges are classified as overlay edges, the global clustering component of the base graph increases, indicating the overlay edges are indeed non-local, leaving local edges responsible for high clustering untouched. This is also seen in the very low clustering coefficient of the overlay graph even for $r_{\text{split}} = 0.9$.

5 Conclusion

We have taken a closer look at the common-neighbors (CN) metric—a metric that forms the basis of many approaches and techniques in outlier detection and graph clustering. Considering a scenario of mixed graphs made up of a base graph with high locality and an overlay graph representing noise, we have shown empirically that the simple CN metric is very accurate and robust for partitioning the edge set, even in the presence of much noise. In addition, the metric can handle real-world networks and partition them into two edge sets of differing properties, helping understand the underlying structures. Our theoretical analysis also gives indications to why the metric works for simple graph models.

A better understanding of this foundational metric is the basis for understanding and designing improved metrics in the fields of outlier detection and graph clustering. We have shown how the metric relates to locality and clustering, and our work indicates interesting related questions. In particular, it would be helpful to further analyze the metric for more complex graph models, including different noise models. In addition, it would be valuable to determine the other factors besides locality that influence the quality of the CN metric, including the degree distribution or density.

References

1. Aggarwal, C.C.: Outlier Detection in Graphs and Networks, pp. 369–397 (2017)
2. Aggarwal, C.C., He, G., Zhao, P.: Edge classification in networks. In: ICDE, pp. 1038–1049 (2016)
3. Albert, R., Barabási, A.L.: Statistical mechanics of complex networks. Rev. Mod. Phys. **74**, 47–97 (2002)
4. Bläsius, T., Fischbeck, P.: On the external validity of average-case analyses of graph algorithms. In: 30th Annual European Symposium on Algorithms (ESA 2022), vol. 244, pp. 21:1–21:14 (2022). https://doi.org/10.4230/LIPIcs.ESA.2022.21
5. Bläsius, T., Fischbeck, P.: On the External Validity of Average-Case Analyses of Graph Algorithms (Data, Docker, and Code), May 2022
6. Chakrabarti, D.: AutoPart: parameter-free graph partitioning and outlier detection. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) PKDD 2004. LNCS (LNAI), vol. 3202, pp. 112–124. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30116-5_13
7. Csardi, G., Nepusz, T.: The igraph software package for complex network research. InterJournal Complex Systems, 1695 (2006)
8. Erdős, P., Rényi, A.: On random graphs I. Publicationes Mathematicae **6**, 290–297 (1959)
9. Hautamaki, V., Karkkainen, I., Franti, P.: Outlier detection using k-nearest neighbour graph. In: ICPR, vol. 3, pp. 430–433 (2004)
10. Kou, Y., Lu, C.T., Dos Santos, R.F.: Spatial outlier detection: a graph-based approach. In: ICTAI, vol. 1, pp. 281–288 (2007)
11. Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A., Boguñá, M.: Hyperbolic geometry of complex networks. Phys. Rev. E **82**, 036106 (2010)
12. Lü, L., Zhou, T.: Link prediction in complex networks: a survey. Physica A **390**(6), 1150–1170 (2011)

13. Mansour, R.F., Abdel-Khalek, S., Hilali-Jaghdam, I., Nebhen, J., Cho, W., Joshi, G.P.: An intelligent outlier detection with machine learning empowered big data analytics for mobile edge computing. Clust. Comput. (2021)
14. Newman, M., Barabási, A., Watts, D.: The Structure and Dynamics of Networks. Princeton Studies in Complexity, Princeton University Press (2011)
15. Pandhre, S., Gupta, M., Balasubramanian, V.N.: Community-based outlier detection for edge-attributed graphs. CoRR abs/1612.09435 (2016)
16. Penrose, M.: Random Geometric Graphs, vol. 5. OUP Oxford (2003)
17. Rossi, R.A., Ahmed, N.K.: The network data repository with interactive graph analytics and visualization. In: AAAI (2015)
18. Schaeffer, S.E.: Graph clustering. Comput. Sci. Rev. **1**(1), 27–64 (2007)
19. Staudt, C.L., Sazonovs, A., Meyerhenke, H.: NetworKit: a tool suite for large-scale complex network analysis (2015)
20. Suri, N.N.R.R., Murty, N.M., Athithan, G.: Outlier Detection: Techniques and Applications. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05127-3
21. Zhang, H., Kiranyaz, S., Gabbouj, M.: Outlier edge detection using random graph generation models and applications. J. Big Data **4**(1), 1–25 (2017). https://doi.org/10.1186/s40537-017-0073-8
22. Zhang, H., Kiranyaz, S., Gabbouj, M.: Data clustering based on community structure in mutual k-nearest neighbor graph. In: TSP, pp. 1–7 (2018)