Multilayer Network Backboning

Valerio Siniscalco
University of Milan
Milan, Italy
valerio.siniscalco@studenti.unimi.it

Michele Coscia
IT University of Copenhagen
Copenhagen, Denmark
mcos@itu.dk

Abstract

Data collection comes with some amount of measurement error and network data is not exempt from this issue. Once a network is collected, it is necessary to test whether an edge exists with some degree of statistical certainty. This is a process called "network backboning." There are a number of approaches to perform network backboning, but the approaches in the literature only work for directed, bipartite networks, and/or hypergraphs. To the best of our knowledge, there is no network backboning approach that takes into account the additional information that comes from multilayer networks. In this paper, we provide the first such approach, by extending the popular noise-corrected backboning approach using a similar Bayesian framework to consider potential layer-layer (anti)correlations. In synthetic experiments, we show that using the additional information that comes from the multiple layers is beneficial to find more accurate network backbones, at a minimal additional time complexity cost. We provide an extensive case study using a network from Wikipedia to show the real world usefulness of extracting multilayer backbones.

CCS Concepts

• **Computing methodologies → Machine learning algorithms**;
• **Applied computing → Mathematics and statistics**.

Keywords

network backboning, network analysis, statistical significance, bayesian framework

ACM Reference Format:
Valerio Siniscalco and Michele Coscia. 2026. Multilayer Network Backboning. In *Proceedings of the 35th ACM International Conference on Information and Knowledge Management (CIKM '26), November 07–11, 2026, Rome, Italy.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3799682.3840676

1 Introduction

Measurement error is the difference between a true quantity of interest and the actual measured value by a sensor. It is a pervasive problem in science and engineering, which prompted the development of a large number of statistical techniques to handle it [40]. Just like any other type of data, network data is subject to measurement error as well. When focusing on edges, we might have the case where some edges we observe are not really present (type I error), while unobserved edges might actually exist (type II error).

CIKM '26, Rome, Italy

ACM ISBN 979-8-4007-2539-5/2026/11
https://doi.org/10.1145/3799682.3840676

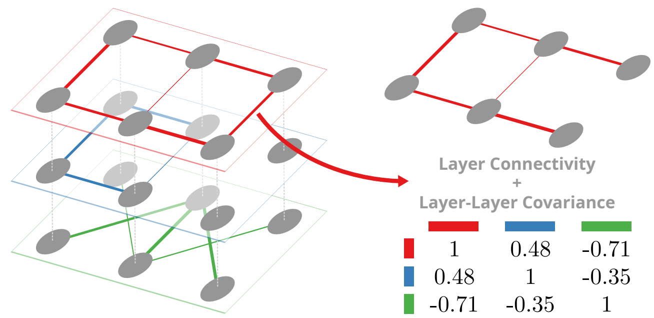


Figure 1: An outline of the multilayer network backboning problem and solution. Given a multilayer network (left: edge colors indicate layers, edge widths indicate the weight), we estimate the probability of an edge existing (at the base of the arrow), which is given by the layer connectivity pattern (right top) and the layer-layer covariances (right bottom).

There are a number of techniques that try to address both error types at the same time [39, 42, 52], while link prediction specializes in dealing with type II errors, inferring the existence of unobserved edges [24, 33, 57].

In this paper, we focus on the network backboning subfield in network error correction, which specializes in type I errors: given an observed noisy weighted network, identify which edges are likely to be spurious [9, 54, 55]. We specifically focus on backboning multilayer networks, by taking into account layer-layer correlations: Figure 1 provides a depiction of the problem we tackle. Network backboning is a particularly popular approach because many real world observed networks are too dense: backboning them results in a sparsification of the graph which, at the same time, reinforces the real signal behind the noisy network data. The resulting network is easier to handle and leads to cleaner results.

Network backboning is as old as network science. The simplest form of backboning – removing from the network all edges with a weight lower than a given threshold – has been a widespread practice since the inception of network analysis. By now, more sophisticated techniques are used for virtually any application of network science: from political polarization [14] to neuroscience [50], from medicine [59] to urban infrastructure [29], including applications such as gastronomy [1], social influence [8], chemistry [26], and art studies [11].

Such a diverse set of applications means that the networks on which we need to apply backboning have all possible sorts of features. For this reason, a wide variety of backboning methods have been developed to take into account the many different topological properties a network can have. For instance, we have backboning methods for directed networks with asymmetric edges; for bipartite networks, containing only edges between two classes of nodes; and hypergraphs, whose edges connect more than two nodes at the same time [10, 35, 36].

Here, we identify an important gap in this literature: multilayer networks. Multilayer networks are networks containing multiple different qualitative types of edges. For instance, you can consider a social network, where different layers represent different modes of interaction between people – face by face, phone calls, texts, social media [4, 22]. To the best of our knowledge, there are no network backboning approaches that target specifically multilayer networks. In this paper, we aim at providing one.

One reason why such a gap in the literature exists could be because one can apply a non-multilayer backboning approach to a multilayer network, by representing it as a multigraph without edge types. Each parallel edge between any node pair can be tested independently, and can be included in the backbone if it passes whichever criterion defined by the backboning approach. The problem is that layers in a multilayer network are not independent from each other. The existence of an edge in one layer influences its probability of existence in another layer. For instance, in a social network, being friends in a face-to-face interaction increases the likelihood we will call each other on the phone – a positive correlation. On the other hand, being connected with a negative type interaction, for instance stealing money from each other, decreases the likelihood of connecting in another, positive layer. Therefore, we cannot have a trivial solution increasing the likelihood of edges simply because they appear in multiple layers.

In our approach, we take inspiration from the Noise-Corrected (NC) backboning algorithm [10]. We do so, because the NC backboning can be extended by updating its priors via the estimation of the layer-layer covariance matrix – which can take into account both positive and negative correlations across layers. Doing so allows us to have a proper multilayer estimation of the likelihood of an edge existing in a specific layer, by integrating information not only from within that layer, but also across layers. This results in a proper Multi-Layer Noise-Corrected (MLNC) backbone.

We evaluate our proposal in two ways. First, we create a synthetic multilayer network model with tunable error rate and layer-layer correlation levels. Here, we have perfect knowledge about which edges are signal and which are noise. We show that MLNC outperforms both the original NC backboning as well as the most used backbone technique in the literature – the Disparity Filter (DF) [46] – in recovering a larger portion of the signal edges. We also show that our approach is scalable.

In our second evaluation we perform an extensive case study with real world data coming from Wikipedia. In the study we show how useful extracting multilayer backbones can be in a cultural data analytics question. Using traditional backbones – as was done in a literature study [23] – provides a contradicting and unclear picture when investigating discrimination in the inclusion of biographies on Wikipedia. On the other hand, our multilayer backboning technique provides a clear and easily-interpretable answer, uncovering no bias in the choices of Wikipedia about whom to include in the encyclopedia's biographies.

In summary, our paper provides the following contributions to the state of the art:

- We are the first to focus on the problem of multilayer network backboning, formally introducing this problem to the literature as a stand-alone problem and not as a side effect of regular network backboning;
- We provide a novel Bayesian framework that can extract multilayer backbones more effectively than any currently available alternative, the only backboning approach taking into account layer-layer correlations;
- We introduce a novel multilayer dataset that can be used for future multilayer backbone benchmarking;
- We advance the field of cultural data analytics by investigating potential discrimination in the inclusion of biographies of Wikipedia, showing that no such discrimination exists.

The data and code on which this paper is based is publicly available and open source[1], which is a further contribution for replicability and for developing future alternative multilayer backbones.

2 Related Works

As mentioned in the introduction, network backboning is one of the oldest problems in network analysis, originally approached in a naive way: to sparsify a network by removing all edges whose weight is below a certain threshold [9]. This is what we can call "naive backboning" which has a number of problems – edge weights distribute broadly and have local correlations: establishing a global fixed threshold risks destroying the network's structure in low-weighted areas and not sparsifying high-weighted areas. In this section we will contextualize the work done in network backboning, and we refer to recent surveys for a more in-depth treatment of the subject [54, 55]. The popularity of network backboning also led researchers to develop software solutions for wider use outside the network science community [37].

The most important and used approaches for classical network backboning – involving simple networks, occasionally directed with asymmetric edges – are the double-stochastic approach based on the normalization of the rows and columns of the weighted adjacency matrix [47], the extraction of the High-Salience Skeleton based on short path trees [15], and other [13, 44, 49, 58]. These are all "structural" approaches that do not consider the statistical significance of an edge weight, but apply an algorithm on the topology of the network to identify which edges are playing an important structural role. It is not trivial to translate these algorithms to multilayer networks, and we opt to use a statistical approach instead.

There are a few classical statistical approaches, which are the most used in network science: the Disparity Filter (DF) [46], the Noise-Corrected [10], and others [32, 43]. The general idea underlying these approaches is the same: to create a null model with an expected edge weight and then test whether the observed edge weight is significantly higher than this expectation. Each model simply differs in how the null model is defined: some like DF use a node-based null model, others like NC use an edge-based null model instead. We identify the gap here: while these models can work on a multigraph representation of a multilayer network, they ignore layer-layer interactions and therefore fall short if we want to define a proper multilayer network backboning approach.

[1] Code: https://www.michelecoscia.com/?page_id=287; Raw data: https://zenodo.org/records/20247946.

This shortcoming has been addressed in different scenarios, for instance for bipartite networks [36, 38, 45]. Bipartite networks need an updated null model, because such a model has to take into account that some edges in a bipartite network are not possible: those that would connect nodes of the same type. Unfortunately, there is not a direct mapping between bipartite and multilayer networks, so a new approach is needed. The same can be said for hierarchical backboning, which updates the null model to be aware that a network might have a hierarchical organization [18]. Hierarchical and bipartite backboning has been combined [19], but still without applicability to multilayer networks.

Another development involves hypergraphs and/or simplicial complexes. Since in hypergraphs and simplicial complexes edges can connect more than two nodes at the same time, we need to extend the null model powering network backboning [31, 35]. While important, these developments are not applicable to multilayer networks. Network backboning has underwent a number of other developments, such as parameter-free approaches driven by the minimum description length principle [20], or a framing as link prediction, to exploit the power of graph neural networks [3, 30].

The closest related work to ours is the introduction of the idea of edges with different types, specifically signed networks [16]. Signed networks are a special case of multilayer networks where edges can be "positive" – like friendship – and "negative" – like enmity. Signs are useful because they increase the predictability of the system: social networks tend to be balanced [2], following intuitive patterns such as the "enemy of my enemy is my friend" which we can call social balance [21, 27] or social status [28]. Crucially, though, backboning approaches in the signed network literature provide signed network as an *output*, not as an *input* (like we intend): if an edge weight is significantly *below* the null expectation it is still returned – not discarded –, with a negative sign. This makes it a completely different problem definition than the one we consider here – which is the classical approach of network backboning of only returning edges significantly *exceeding* null expectation.

Some work has been done for signed bipartite networks [12] and for graph generation [7]. We integrate some of these ideas in our work, but ultimately these related work fall short because they do not develop an actual multilayer backboning approach.

3 Methodology

3.1 Problem Definition

We start by establishing some notation. Let $G = (V, E, L)$ be a directed multilayer graph, composed by a node set V, an edge set E, and the layer set L. We define each edge in E to be a quadruple (u, v, l, w), with $u, v \in V$ being the two nodes connected by an edge, through the layer $l \in L$, and with weight $w \in \mathbb{Z}^+$ being a positive integer. Note that we can allow for any arbitrary positive real weight – i.e. $w \in \mathbb{R}^+$ – by simply replacing in Section 3.2 the Binomial function with an appropriate continuous function. For simplicity – and consistency with previous work – we limit ourselves to discrete edge weights.

Since G is directed, $(u, v, l, w_{uvl}) \neq (v, u, l, w_{vul})$, i.e. nodes u and v in the same layer l might have different weights w_{uvl} and w_{vul}. In fact, it is possible – and common – that even if (u, v, l, w_{uvl}) exists, (v, u, l, w_{vul}) does not – i.e. $w_{vul} = 0$ for our purposes.

In this paper we assume that each edge is measured with a certain amount of error. We cannot therefore be sure that a measured edge exists and we need a way to estimate the likelihood of an edge in a layer to have actual weight of zero – i.e. it does not exist. This leads us to our formal problem definition:

Definition 1. *Given a multilayer graph $G = (V, E, L)$ with discrete count layered edge weights defined as (u, v, l, w), define a function $f : G, (u, v, l, w) \rightarrow [0, 1]$ such that the output represents the likelihood of edge (u, v, l, w) to have a weight $w > 0$.*

Note that this problem definition acknowledges the multilayer aspect of the graph, but does not necessarily require that this information is taken into account. This is because classical single-layer backboning methods can work on multilayer graphs, by simply ignoring the layer information coming from L. The objective of this paper is proving that doing so results in an inaccurate output, and less useful insights in real world scenarios.

3.2 Single Layer Backboning

We now outline the Noise-Corrected (NC) backboning approach in a single layer setting [10]. Doing so is useful, as our proposed multilayer approach is a direct evolution of this single-layer variant.

NC backboning uses a weighted configuration model (CM) expectation for each edge in the network, and then tests the likelihood of the observed edge weight to be greater than zero by comparing it against a binomial null extraction process. The weighted CM estimates the probability of any (u, v) edge to be:

$$E[p_{uv}] = \frac{w_{u.} w_{.v}}{w_{..}^2},$$

with $w_{u.} = \sum_v w_{uv}$ being the total outgoing edge weight coming from node u – and, likewise, $w_{.v} = \sum_u w_{uv}$ being the total incoming edge weight pointing towards node v –, and $w_{..} = \sum_{u,v} w_{uv}$ being the total edge weight mass of the entire graph G. The variance $V[p_{uv}]$ is estimated via a Bayesian framework – here omitted for brevity, see [10] for the full derivation – whose objective is to avoid being over-confident about low-degree nodes – which are common in most real-world networks.

Ultimately, we obtain a null $\tilde{p}_{uv}$ probability of connecting nodes u and v, which we can plug in a binomial approximation to model the expected edge weight between the two nodes: $Binomial(w_{..}, \tilde{p}_{uv})$. This allows us to define our function f estimating the likelihood of the edge to surpass this null expectation:

$$\begin{aligned} f(u, v, w_{uv}) &= \mathbb{P}(X < w_{uv}) \\ &= f_{Binomial}(w_{uv}; w_{..}, \tilde{p}_{uv}) \\ &= \sum_{i=0}^{w_{uv}-1} \binom{w_{..}}{i} \tilde{p}_{uv}^i (1 - \tilde{p}_{uv})^{n-i}. \end{aligned}$$

One can see how it is trivial to extend this methodology to fit our problem definition. Two edges connecting the same u, v pair in different layers l_1 and l_2 still contribute to the $w_{u.}$ and $w_{.v}$ sums, leading this multilayer information to be taken into account in estimating $\tilde{p}_{uv}$. Moreover, since it is possible that $w_{uvl_1} \neq w_{uvl_2}$, then the two edges in the two different layers will obtain different f scores, making the classical NC backbone approach a multilayer one. This holds for all similar statistical backbone approaches such as the disparity filter [46]. However, this operation rests on two strong assumptions: it considers layers to be (i) *independent* and (ii) *indistinguishable* from one another.

Layers are considered (i) *independent* because any potential (anti)correlation between edges in different layers is ignored: only the weight w_{uv} is considered in estimating $\tilde{p}_{uv}$. However, this is not the case in real-world multilayer networks: in a multilayer social network with a different social media per layer, we expect friendships to be correlated – it is more likely for u and v to be friends on Facebook if they are also friends in Bluesky. Or in a social game, it is less likely to be connected in the friendship layer if u and v attacked each other in the game, a connection in a conflict layer.

Furthermore, layers are also considered (ii) *indistinguishable* because $w_{..}$ pools information from all of them at the same time and each layer tests itself against a shared null model. A side effect of this choice results in the fact that the null model ignores that some edges might connect more or less strongly than others in general. Suppose that the average edge weight is much lower in l_1 than in l_2. Then, for any pair of u, v nodes connected in both layers, the backbone will likely consider as significant only edges from l_2, de-facto creating an almost empty backbone for layer l_1. For these reasons, we need to estimate layer-dependent $w..l$ and $\tilde{p}_{uvl}$.

3.3 Multi Layer Backboning

The first step to address the shortcomings of the layer-agnostic backbone extraction is to disaggregate information by layer. We therefore move to a per-layer expected connection probability:

$$E[p_{uvl}] = \frac{w_{u.l} w_{.vl}}{w_{..l}^2},$$

with the trivial extensions from the baseline: $w_{u.l} = \sum_v w_{uvl}$, $w_{.vl} = \sum_u w_{uvl}$, and $w_{..l} = \sum_{u,v} w_{uvl}$.

For convenience, we work with log-odds rather than directly with probabilities: $x_{uvl} = \text{logit}\,(E[p_{uvl}])$. The assumption in our model is that the odds of nodes u and v to connect in layer l are the sum of their baseline odds of connecting in general (μ_{uv}), plus a layer-specific factor ℓ_{uvl}:

$$x_{uvl} = \mu_{uv} + \ell_{uvl}.$$

With x_{uvl} we can *distinguish* between layers, addressing shortcoming (ii), by effectively giving each layer its own configuration models. These configuration models are, however, still *independent*. To estimate the dependencies between them we need to calculate the layer-layer covariance matrix Σ:

$$\Sigma_{l_1,l_2} = (|E| - 1)^{-1} \sum_{u,v} \left(x_{uvl_1} - \bar{x}_{uv}\right)\left(x_{uvl_2} - \bar{x}_{uv}\right),$$

with $\bar{x}_{uv} = |L|^{-1} \sum_l x_{uvl}$ being the average connection strength between nodes u and v across all layers.

This seemingly innocuous step actually requires attention, because in many cases an edge will not appear in all layers – especially for anti-correlated layers. We therefore need a protocol to deal with missing x_{uvl} values. There are two alternatives: either the values are treated as true missing and therefore ignored in the average calculation, or they are treated as measured zeroes and therefore we impute them as $x_{uvl} = \text{logit}\,(\epsilon)$, with ϵ being an arbitrarily low value – necessary because we cannot take the logit of zero. The choice between these two options should be left to the user, as they know whether missing connections in a layer are unobserved edges, or observed zero-weighted edges.

We now provide the posterior mean μ_{uv}, which estimates the affinity of the edge across the layers, by normalizing the observed odds of the edge existing with the prior of the edge not existing – shifting it towards zero:

$$\tilde{\mu}_{uv} = \frac{\mathbf{1}^T \Sigma^{-1} x_{uv}}{\mathbf{1}^T \Sigma^{-1} \mathbf{1} + \frac{1}{\tau^2}},$$

where $\mathbf{1}$ is a vector of ones and τ is the parameter controlling the global pooling strength. High values of τ should be used when we expect the layers to have a strong shared signal, vice versa τ should be set to a low value when we believe the layers to be largely independent. For typical cases, $\tau = 1$.

We then perform the layer-wise shrinkage using our posterior of the edge affinity across layers, which gives us our final null expectation of the edge's odds of existing in a given layer:

$$\tilde{x}_{uvl} = \tilde{\mu}_{uv} + \lambda \left(x_{uvl} - \tilde{\mu}_{uv}\right).$$

Here, λ is the second parameter of our framework, which regulates the shrinkage strength. For low $\lambda = 0.1$ we force the layers to be similar (correlated) with the estimated $\tilde{\mu}_{uv}$ overall edge affinity; for high $\lambda = 0.9$ we result in layers largely independent from $\tilde{\mu}_{uv}$. Typically, $\lambda = 0.5$.

At this point we can go back to work with probability reverting the logit transformation, $\tilde{p}_{uvl} = \text{logit}^{-1}\,(\tilde{x}_{uvl})$, and use it to perform the same null model test we use in the single layer case:

$$\begin{aligned} f(u, v, l, w_{uvl}) &= \mathbb{P}\,(X < w_{uvl}) \\ &= f_{Binomial}(w_{uvl}; w_{..l}, \tilde{p}_{uvl}) \\ &= \sum_{i=0}^{w_{uvl}-1} \binom{w_{..l}}{i} \tilde{p}_{uvl}^i (1 - \tilde{p}_{uvl})^{n-i}, \end{aligned}$$

but now we have a proper integration of the information coming from the layers. Each layer l is a parameter of function f, and $\tilde{p}_{uvl}$ includes the information of the (anti)correlation between layers.

4 Experiments

We evaluate our multilayer backboning in two ways. In this section, we leverage synthetic data to validate it (Section 4.1) and discuss its computational complexity (Section 4.2). In Section 5 we create a novel multilayer network dataset from Wikipedia and show the usefulness of multilayer backboning for cultural data analytics.

4.1 Validation

To validate our multilayer backboning we generate a synthetic network using a Stochastic Blockmodel (SBM) approach. There are a few examples of multilayer SBMs in the literature [17, 41, 48, 51]. Here we use a custom multilayer SBM in which we can tune the amount of noise in the edges via a parameter η.

We fix a layer-layer correlation via a parameter $\rho \in [-1, +1]$. When $\rho < 0$, edges that appear in one layer tend not to appear in

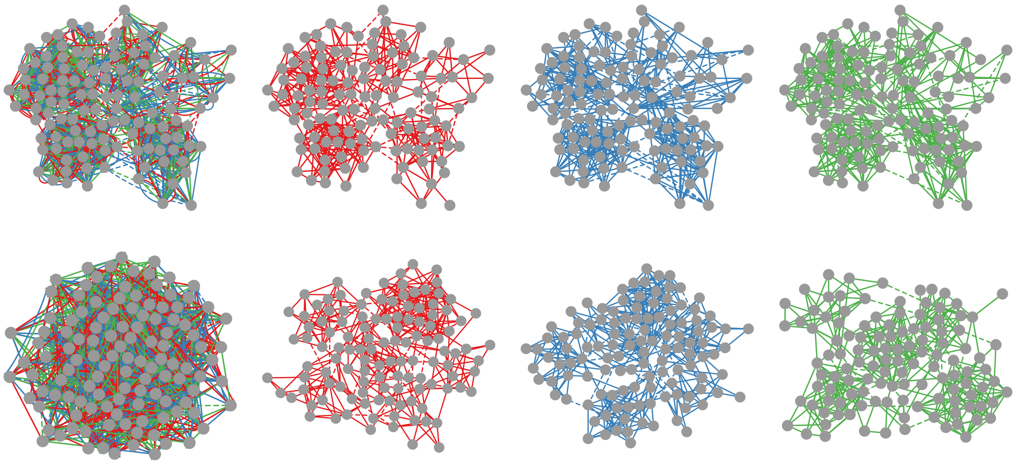


Figure 2: Two samples of noisy multilayer SBMs. Edge colors represent the layer in which an edge appear. Dashed edges are drawn from noise, solid edges are signal. Above row for high positive ρ (layer-layer correlation), row below for high negative ρ (anti-correlation).

other layers, vice versa if ρ is positive. If $\rho = 0$ then an edge in one layer does not influence its appearance in other layers. We set a number of communities, and ρ regulates how aligned across layers the communities are. With probability proportional to p_{in} we draw an edge between nodes in the same community, and with $p_{out} << p_{in}$ we draw an edge between nodes in different communities. Each edge has a random weight drawn from a Poisson distribution.

Note that p_{in} and p_{out} are not the actual probabilities of connecting: they still need to be drawn via a multivariate normal process ruled by a covariance matrix Σ, which is in turn initialized by ρ. This is because even if nodes u and v are drawn in the same community in two layers l_1 and l_2, if $\rho < 1$ then we still need to make sure to respect the level of correlation between edges in l_1 and l_2.

Once the signal edges are extracted, we add a certain amount of noise using η. If a u, v node pair was not connected by the SBM, we have an η probability of connecting them via a noisy edge. Noisy edges have weights that are drawn from the same Poisson distribution as non-noisy edges. This means that, within a layer, noisy edges are somewhat indistinguishable from non-noisy edges. However, because they are embedded in a multilayer structure, a multilayer backboning should be able to identify them.

Figure 2 shows examples of the multilayer synthetic networks we generate at different levels of ρ, which confirms visually that our noisy multilayer SBM is generating a proper test set: for high positive ρ the community structure is noticeable even when all layers are plotted at the same time (note that in this row each node has the same x,y coordinate in all subfigures). For high negative ρ, while the layer-wise community structure is noticeable in each layer in isolation, since the layers are anti-correlated the global network shows no community structure.

To validate the backboning methods we use the Jaccard coefficient between the edges recovered in the backbone and the non-noisy edges in the original structure. If E_{bb} is the set of edges returned by the backbone and E_{real} is the set of the real edges, then our performance is:

$$perf_{bb} = \frac{|E_{bb} \cap E_{real}|}{|E_{bb} \cup E_{real}|}.$$

We test our proposed Multi-Layer Noise-Corrected (MLNC) method against the single layer state of the art, since there are no other multilayer backboning methods and, as we saw in Section 3.2, a statistically-based single layer backboning method can give multilayer p-values. We choose to compare against single layer Noise-Corrected (NC) [10], since our model is a direct evolution of the same idea, as well as the Disparity Filter (DF) [46], since this is by far the most used statistical backbone extraction method. Structural extraction methods cannot be compared, since their multilayer extension is not straightforward. We also include naive thresholding, to contextualize what would happen if we were to pick only the strongest edges ignoring layer information.

We repeat each test for any given parameter combination 20 times and report the average plus/minus the standard deviation.

Figure 3 shows the performance of these three methods on a number of tests we run, each varying some parameters in the noisy multilayer SBM generation. First, we vary the noise parameter η (Figure 3a). As expected, the larger the noise η the harder it is to recover the real structure. It is clear from the figure that taking into account the information shared across layers in the MLNC backbone is useful to retain as much signal as possible. The gap in performance is evident. Interestingly, all non-multilayer alternatives perform at a remarkably similar level, including Naive thresholding, showing how harmful it is even for sophisticated backboning methods to ignore the information coming from the layers.

Figure 3b shows what happens when we have different levels of correlations between layers – here we fix $\eta = 0.02$. While there seems to be a small positive effect of layer correlation (for DF we go from $\sim 0.56 \rightarrow\sim 0.6$ and for MLNC $\sim 0.71 \rightarrow 0.72$ when moving from strong anti correlation to strong positive correlation) this is not statistically significant. Therefore, none of the backboning methods are affected, crucially including our approach MLNC. This means that MLNC does not have an edge only when in strong

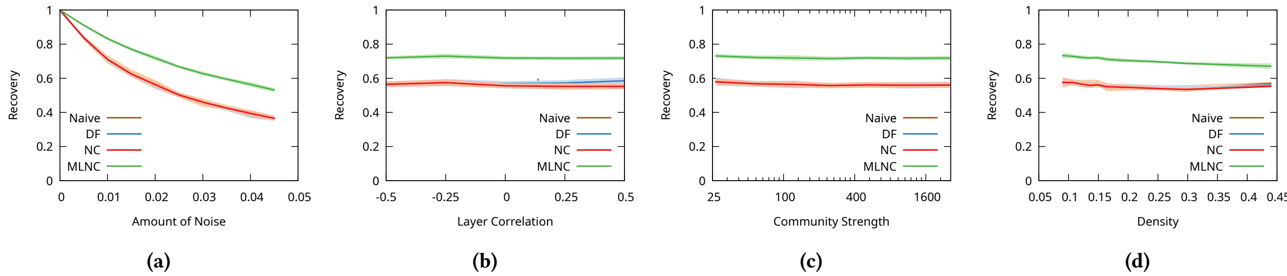


Figure 3: The backbone performance (y axis) varying different parameters in the SBM (x axis): (a) level of noise η, (b) layer-layer correlation ρ, (c) community strength p_{in}/p_{out}, (d) network density. The line color identifies the backboning method: brown for Naive, red for NC, blue for DF, green for our method MLNC. The solid line reports the average performance, while the shaded area shows the standard deviation across all runs. Note: only NC is visible among the alternatives due to near identical performance.

(anti)correlations between layers. Even if $\rho = 0.0$, it is still useful to use a multilayer approach as there is still information that can be used to recover the best possible backbone.

The performance of the backboning method is also not influenced by how strong the communities are (Figure 3c). We vary the strength of the communities by changing the p_{in}/p_{out} ratio, making sure the network retains the same overall edge density. Again, there seems to be a non-statistically significant preference for weaker communities (DF $\sim$ 0.56 $\rightarrow\sim$ 0.58 and MLNC $\sim$ 0.72 $\rightarrow\sim$ 0.74, when going from the strongest to the weakest communities).

Density (Figure 3d) presents a more interesting case. All backboning methods struggle with higher densities (e.g. DF $\sim$ 0.57 $\rightarrow\sim$ 0.55). However, MLNC's performance declines more sharply, while remaining still significantly better than the alternatives: $\sim$ 0.72 $\rightarrow\sim$ 0.67. This is to be expected, as we increase the density uniformly between noise and signal edges, therefore there are more noisy edges in denser networks, which might make it harder to identify them. Ultimately, the density values for which MLNC struggles the most are significantly larger than density values that are normally observed in real world networks.

The near-identical performance across all test between DF, NC, and Naive could lead to think these methods return almost the exact same set of edges. Surprisingly, that is not the case. Figure 4 shows the pairwise Jaccard coefficients of all methods – including the real edge signal – at the highest noise level 0.045 – we use the highest noise because obviously at low noise level all backbones tend to overlap with the real signal. For this test, we report the average of 100 independent runs.

The figure confirms that MLNC has the highest overlap with the edge signal. The figure also shows that DF is quite similar to Naive. Even so, the average Jaccard of these two methods is far from indicating a near identity: it is 0.863, high but leaving space for significant disagreement. In networks with one thousand edges, at this level of Jaccard agreement the two methods still return each 100 edges that the other does not. The NC backbone is more distinct still, its Jaccard coefficient with the other two methods hovering around 0.6. This implies that these backbones are not all returning the same result, only that the different results they return are equally uncorrelated with the real signal.

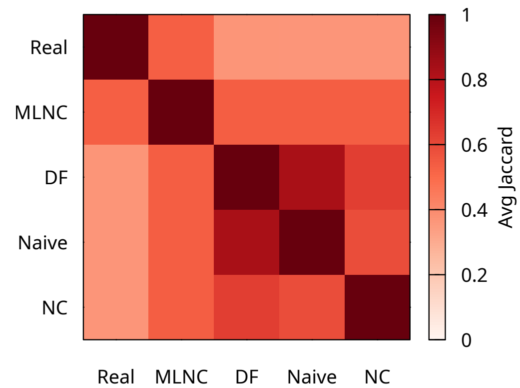


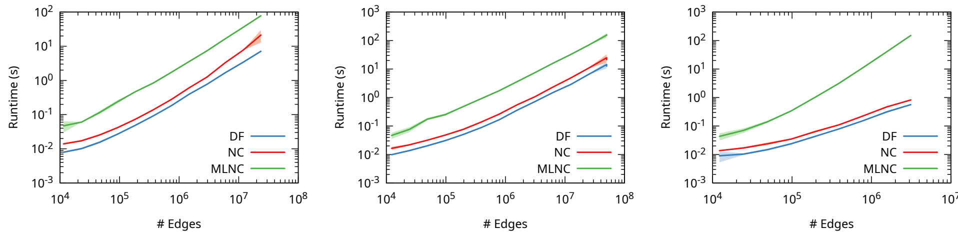
Figure 4: Heatmap of the average pairwise Jaccard coefficient of all backbone methods. Darker color means more agreement. Average of 100 independent runs.

4.2 Efficiency

While we did not focus on providing the most efficient possible implementation of MLNC, we note how all operations of statistical backboning like DF, NC, and MLNC are vectorized and roughly linear in the number of edges. Without a formal derivation, the time complexity of MLNC should be in the order of $O\left(|E||L|^2\right)$, because that is the time complexity of estimating the covariance matrix. This is convenient, as for the vast majority of real-world multilayer networks $|L| << |E|$. The alternatives, DF and NC, being blind to the layers, only see an increase in number of edges regardless of $|L|$, and therefore their time complexity is $O\left(|E|\right)$.

To show the validity of our estimation, we test all three backbones on networks of increasing sizes in terms on $|E|$. For this test, we generate two-layer networks by using a simple $G_{n,p}$ Erdos-Renyi random graph, as we do not really care about the actual backbone, only how much time it takes to compute.

There are three ways to increase the edge set: by adding nodes, or by increasing the edge density in the network, or by adding more layers. We perform all tests, to verify whether there is a difference



(a) (b) (c)

Figure 5: The running time in seconds (y axis) against the network size in number of edges $|E|$ (x axis). The size is regulated by increasing: (a) the node count $|V|$, (b) the average degree, or (c) the number of layers $|L|$. The line color identifies each method: red for NC, blue for DF, green for our method MLNC. The solid line reports the average runtime, while the shaded area shows the standard deviation across all runs.

in increasing $|V|$ as well as $|E|$, in having very dense layers, or simply many of them. For each given network size, we repeat the process ten times and report average and standard deviation of the runtime, to smooth out potential fluctuations.

Figure 5 shows the result. Understandably, MLNC has a worse multiplicative constant runtime when compared with DF and NC, because it needs to take additional steps, evaluating the layer-layer correlations. Besides this constant penalty, the overall time complexity asymptote when increasing node count or network density is the same across all models – as Figures 5a and 5b. In both cases, the best fit for MLNC time complexity is a power function with exponent one, i.e. a linear increase.

Figure 5c confirms this, showing that increasing the layer count does not affect DF and NC – as they are blind to them – but requires additional computation time in MLNC, because MLNC needs to build the layer-layer covariance matrix and use it. In this case, empirically the best fit for MLNC time complexity is a power function $O\left(|E|^{1.44}\right)$. While asymptotically MLNC has worse time complexity in this regard, the total runtime is still manageable even for very large networks with $|L| = 128$ (other parameters being set at $|V| = 2000$ and $|E| = 780140$). As we noted at the beginning of the section, it is quite rare to encounter a multilayer network with more than 128 layers. In fact, we have to stop our test at the size we stop at because the limiting factor taking most of the runtime is generating the random network, and not backboning it.

5 Case Study

We now turn to the case study of the paper: the usage of backboning methods to find discriminatory behavior in the inclusion of notable people on Wikipedia. As one work in the literature showed [23], backboning is useful in this case because by using aggressive thresholds we can drop nodes from the network in a specific order: from the individuals who have statistically weak connections to those who have statistically strong ones. If a certain class of people – say women vs men – tends to be preserved more, it means these people tend to have statistically stronger connections, which implies there is a higher barrier of entry for that class of people.

5.1 Dataset Building

We use this section to document the dataset building procedure, since it represents an important contribution of the paper: not only we provide a valuable novel multilayer network for benchmarking multilayer backboning methods, but we can also use it to provide a contribution to the field of cultural data analytics by investigating biases in inclusion in databases related to history. Wikipedia is de-facto one of the most important sources of information about history for the wider population, and biases in its editorial practices should be investigated.

We collected the data on January 2nd, 2026 using the official API[2], essentially creating a snapshot of the encyclopedia as it appeared on that day. Each node in our V set is an illustrious person with a biography page on Wikipedia. The API allows to obtain a unique ID for each person, regardless of different localizations of their name in different language editions of Wikipedia. Therefore, the nodes across layers are automatically aligned.

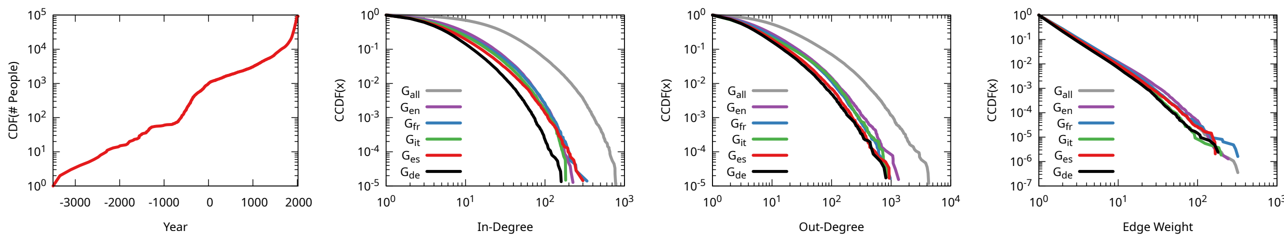
Our first advancement over the state of the art is that we do not narrow our study to individuals born between 1750 and 1950, as is done in the work we improve upon [23]. Moreover, [23] uses Pantheon [6, 56] to identify a curated list of biographies on Wikipedia, but we rely instead on a larger, more comprehensive dataset [25].

Another improvement in our dataset is on the node attributes. We include not only gender as is done in [23], but also the period of birth and the world region encoding the place of birth. This allows us to perform a validation of the methodology using time in Section 5.2, but also to explore Western versus non-Western bias in Section 5.4, besides analyzing gender bias in Section 5.3.

Two individuals u, v are connected by a directed edge if the Wikipedia biography of u links to v's Wikipedia biography. The third improvement we make over [23] is in moving from a single layer to a multilayer network. Originally, only links in the English language Wikipedia were used. Here we add hyperlinks from the French, German, Italian, and Spanish language editions. Each language represents a different layer l in the multilayer network.

Following [23], we remove links that do not come in the body of text of the biography, as they are normally found in footer tables and represent loose and not relevant connections – for instance connecting Aristotle to Rosalind Hursthouse just because the latter

[2] https://www.mediawiki.org/wiki/Wikimedia_APIs



(a) CDF($|V|$) **(b) CCDF(k_{in})** **(c) CCDF(k_{out})** **(d) CCDF(w)**

Figure 6: Cumulative distributions (y axis) of some network variables (x axis): (a) Number of nodes, (b) in-degree, (c) out-degree, (d) edge weight. The line color refers to a different layer, gray for the overall multilayer network aggregating all layers.

is included in a footer table on Aristotle's biography as being one of his followers. The weight of an edge represents the number of times v has been mentioned in u's biography, which we take as proportional to the strength of their relationship.

Unfortunately, not all occurrences are marked with a hyperlink, so we need to employ a variety of natural language processing techniques – specifically DBPedia Spotlight [34] with multilingual support [53] (we set its confidence parameter to 0.8) – to add weights to the edges. There is a serious issue in case of people sharing surnames. Consider the case of reading Mary Shelley's Wikipedia page. Since she does have a hyperlink pointing to Percy Shelley, each occurrence of standalone "Shelley" can be added to the edge weight – even if it was referring to Mary rather than Percy. We can identify other potential spurious edges even without this substring matching issue, by looking at the z-score of each outgoing edge weight from a node when compared with the node's expected outgoing edge weight. We set a threshold to 1.5, identifying 29,169 edges affected by this issue (around 1% of all the edges). Note that the z-score is only applied to the ambiguous, surname-matching pairs – otherwise it would produce a massive number of false positives.

When we find such ambiguity, we fix it with a two-step strategy. First, we switch to a full string match, only adding to the edge weight if we encounter the full name "Percy Shelley" and not just "Shelley", under the assumption that the article writer would write the full name when referring to someone else than the subject of the biography they are writing. Using this strategy, we can add proper weights to 28,312 out of 29,169 edges (97% of the ambiguous edges). However, there are 857 cases in which this fails, because the full name match does not help us: full matching the string "Napoleon" still does not help when looking at the edge of the first emperor of France when reading the biography of his nephew "Napoleon III". The existence of these 857 edges is not in doubt – because the hyperlink exists – but we have no other choice than to put the edge weight to one in the second step of our solution.

Table 1 reports summary statistics about the dataset, showing that it has a skewed weight distribution, with low edge weight average and most edge weights being equal to one. In all layers, the majority of nodes are in the same connected component, whose average path length is between 5 and 7, depending on the layer.

Figure 6 shows some distributions to make better sense of the dataset. Figure 6a shows the cumulative count of nodes as we move through history in our dataset. The figure shows a great growth in number of nodes that continues even after 1950, as well as a significant number of nodes before 1750, which were excluded in [23]. The in- and out-degree distributions (Figures 6b and 6c) show a broad distribution common across layers, which fails to be a power-law. The edge weight distributions (Figure 6d) are remarkably similar across layers. While we do not properly test for power-laws [5], it seems that the edge weights distribute much more broadly than the degree, which is one key characteristic addressed by the noise-corrected style of backboning we deploy here.

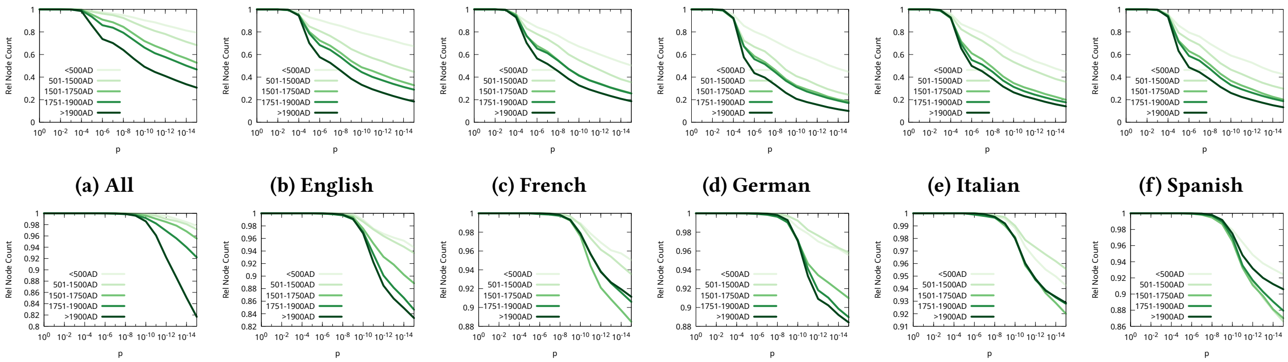
Measure	G_{all}	G_{en}	G_{de}	G_{fr}	G_{it}	G_{es}
$\lvert V\rvert$	92,238	83,823	81,377	79,176	75,591	77,569
$\lvert E\rvert$	2,836,525	776,770	413,514	620,310	542,720	483,211
Avg. w	1.61	1.68	1.48	1.68	1.57	1.58
% $w = 1$	75.11	74.40	78.79	72.88	75.18	75.91
% in LCC	99.59	98.41	97.28	97.74	96.71	96.25
APL	4.99	5.63	6.97	5.82	6.09	6.22

Table 1: The basic statistics of the multilayer network we extract from Wikipedia, and for each of its layers: $|V|$ = number of nodes, $|E|$ = number of edges, Avg. w = average edge weight, % w = 1 = share of edges with weight equal to one, % in LCC = share of nodes in the largest weakly connected component, APL = average path length.

5.2 Period Bias

The first dimension of bias in inclusion we explore is temporal. We divide nodes in five temporal classes based on their year of birth, from antiquity (born before 500BC) to contemporary (born after 1900AD). We consider this as a validation of our methodology. We do not think there necessarily is a conscious bias in excluding people from a particular age, but there will be other sources of biases that makes it harder for people in older eras to be included, e.g record loss over time. We expect to see a strong discriminatory signal against people the further back in the past they were born, as the further we go back in time the more notable a person has to be to find enough records to be included.

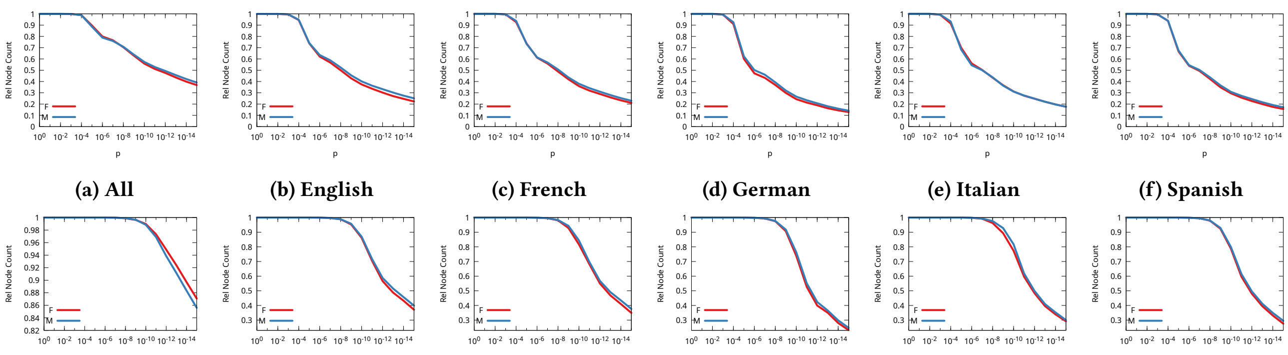
Figure 7 shows the results, for both MLNC and NC – the latter was used in the original paper. For MLNC (above row) we can see that the validation is completely successful: in the multilayer structure and in each of the layers we see the periods sorting themselves correctly. The top row of the figure shows that the further back in the past we go, the higher levels of "discrimination" we observe –



(a) All (b) English (c) French (d) German (e) Italian (f) Spanish

(g) All (h) English (i) French (j) German (k) Italian (l) Spanish

Figure 7: The share of nodes retained in the backbone (y axis) per given p-value threshold (x axis). The line color indicates the period the node is from, from light (ancient) to dark (contemporary). Above row for MLNC, below row for NC.



(a) All (b) English (c) French (d) German (e) Italian (f) Spanish

(g) All (h) English (i) French (j) German (k) Italian (l) Spanish

Figure 8: The share of nodes retained in the backbone (y axis) per given p-value threshold (x axis). The line color indicates the gender of the node: male (blue) or female (red). Above row for MLNC, below row for NC.

more nodes have more significant connections, a sign they needed to clear a higher significance bar to be included.

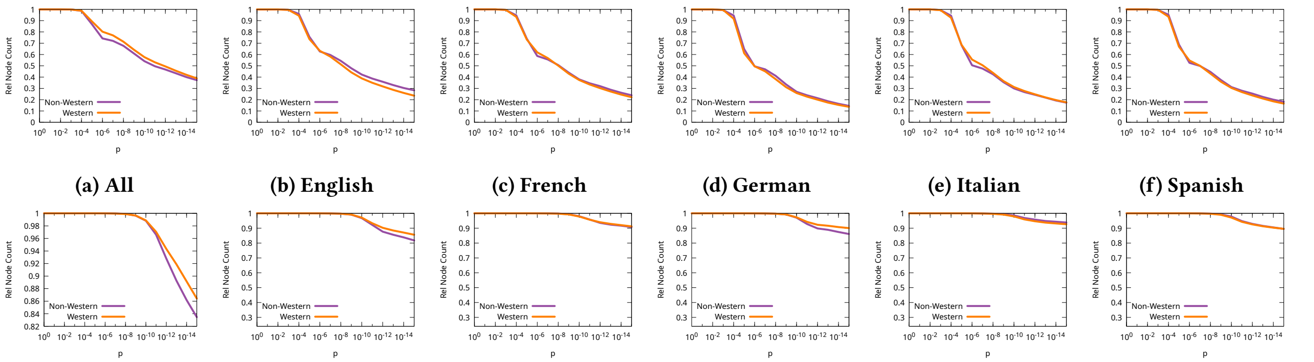
The second row of Figure 7 shows the picture from NC. While the validation is mostly successful in this case, we see some puzzling inconsistencies. In the French and Spanish networks, the second most "discriminated" class is the most recently born people, which we would expect to be the least discriminated as it involves the most recent historic records. The second most recent class (1751AD-1900AD) is also more discriminated that the previous one (1501AD-1750AD), and there are also some out-of-order cases in the Italian and German layers. Worryingly, it is not clear how these inconsistencies translate into the full multilayer structure which, as we can observe, shows none of these surprising patterns. Our working hypothesis is that the overall structure is simply dominated by the largest layer – English – which is also showing the expected sorting across periods. This means that NC, by ignoring the multilayer information, ends up dominated by the largest layer.

5.3 Gender Bias

We now move to a study of gender bias. Figure 8 shows the number of retained nodes per layer at increasing significance thresholds. In [23], a significant gap between female and male nodes had been found, by using the NC backboning. Figure 8g confirms this finding when using the NC backboning: at higher significant threshold, more female nodes retain their place in the overall multilayer structure, by preserving at least one connection.

However, when analyzing the structure layer by layer – in Figures 8h to 8l – a puzzle emerges. This gap between male and female nodes is nowhere to be found in any of the layers. The number of retained nodes is exactly the same across the two genders – slightly higher for males, in fact. This creates a situation that is difficult to explain: the overall multilayer structure shows a pattern that is not present in any of the layers. The conclusion from this contradicting analysis would be that Wikipedia overall – across language editions – is biased, but none of the language editions are.

If we instead use our proposed MLNC approach, this contradiction disappears. Figure 8a shows that indeed the overall multilayer structure is not biased against female nodes, and this pattern is



(a) All (b) English (c) French (d) German (e) Italian (f) Spanish

(g) All (h) English (i) French (j) German (k) Italian (l) Spanish

Figure 9: The share of nodes retained in the backbone (y axis) per given p-value threshold (x axis). The line color indicates the region the node is from: western (orange) or non-western (purple). Above row for MLNC, below row for NC.

confirmed in all language layers. Figures 8b to 8f consistently show no discriminatory pattern, creating a clear picture.

Our conclusion is that Wikipedia does not add any significant reporting bias against women. There are still a number of biases against women – historic lack of access, recognition, as well as biases in historiography, clearly evident from the fact that only 18.4% of profiles in out database are women – but Wikipedia is not adding additional biases on top of these.

5.4 Origin Bias

For the bias in the origins of illustrious people, we divide the Wikipedia profiles in two classes depending on their place of birth. People born in Western, Southern, or Northern Europe, the Americas, or Australia/New Zealand are labeled as "Western," and everybody else as "Non-Western." Then we perform the same analysis we did for gender. Figure 9 displays the results.

Our analysis on origin bias fortifies the unsuitability of ignoring layer information, which we already discussed for gender. We see again a contradiction in the NC results between the pattern of discrimination we observe in the overall multilayer structure – Figure 9g – and the absence of such pattern in all of the layers composing it – Figures 9h to 9l. This contradiction is made worse by the fact that the discrimination we observe in the main multilayer network shows a bias against Westerners, which seems unlikely.

Once more, MLNC fixes this inconsistency. No discrimination pattern can be found in the overall multilayer network – Figure 9a –, which is consistent with the absence of discrimination in each of the language editions – Figures 9b to 9f. Again, we remind the reader that this means Wikipedia does not add any additional bias to the ones already present in either history or historiography, which provide the sources for Wikipedia itself. Non-Western people in our dataset only include 20.1% of the profiles, despite their areas of origin providing a much higher share of historic population. We also should not over-interpret these results, as our cultural origin classification is superficial, and requires future work in cultural data analytics to be made more accurate.

6 Conclusion

In this paper, we tackled the issue of creating a multilayer backboning approach. Backboning is the task of identifying significant connections in a noisy complex network. This task has attracted great interest in the literature, which has proposed methods to extract backbones in directed, bipartite, and hyper networks. Our approach is the first one targeting multilayer networks. In synthetic experiments, we show how our approach is better suited for this specific data type, being able to recover more efficiently signal edges over noisy ones. We deployed our MLNC approach on a case study on cultural data analytics: the analysis of inclusion bias in the biographies of notable people on Wikipedia. Our approach showed its validity in correctly identifying period bias. When analyzing gender and origin bias, we fail to find a systematic Wikipedia-induced bias. The usefulness of our approach is in providing a clean, consistent signal, which moves beyond what has been done in the literature: non-multilayer approaches provide a confusing, contradictory picture.

There are a number of potential future works opened by our work. First, we have targeted a specific backboning algorithm, and extended it to the multilayer case. There is work to be done exploring the multilayer extensions of other backboning approaches. Relatedly, our chosen approach is statistical, meaning that we create a null model to compare against the observation. However, there are a number of structural backboning approaches, whose multilayer extensions could require quite different solutions from the strategy we deploy here. Finally, our cultural data analytics case study also requires further development. Our analysis of origin bias – comparing how Wikipedia treats Westerners and Non-Westerners – is superficial in nature. We need a more robust node classification if we want to properly talk about inclusion biases in Wikipedia.

Acknowledgments

We thank the Danish Data Science Academy for sponsoring Siniscalco's travel grant, which made this collaboration possible; and Frank Neffke for insightful discussions.

GenAI Usage Disclosure

No Generative AI has been used in the design of this study nor for writing this manuscript. Generative AI has been used for documentation and code revisions.

References

[1] Yong-Yeol Ahn, Sebastian E Ahnert, James P Bagrow, and Albert-László Barabási. 2011. Flavor network and the principles of food pairing. *Scientific reports* 1, 1 (2011), 196.

[2] Samin Aref and Mark C Wilson. 2019. Balance and frustration in signed networks. *Journal of Complex Networks* 7, 2 (2019), 163–189.

[3] Jason Barbour, Ali Yassin, Hocine Cherifi, and Hamida Seba. 2024. Backbone extraction in complex networks via embedding-based link prediction. In *International Conference on Complex Networks and Their Applications*. Springer, 224–236.

[4] Stefano Boccaletti, Ginestra Bianconi, Regino Criado, Charo I Del Genio, Jesús Gómez-Gardenes, Miguel Romance, Irene Sendina-Nadal, Zhen Wang, and Massimiliano Zanin. 2014. The structure and dynamics of multilayer networks. *Physics reports* 544, 1 (2014), 1–122.

[5] Anna D Broido and Aaron Clauset. 2019. Scale-free networks are rare. *Nature communications* 10, 1 (2019), 1017.

[6] Cristian Candia, C Jara-Figueroa, Carlos Rodriguez-Sickert, Albert-László Barabási, and César A Hidalgo. 2019. The universal decay of collective memory and attention. *Nature human behaviour* 3, 1 (2019), 82–91.

[7] Giona Casiraghi and Vahan Nanumyan. 2021. Configuration models as an urn problem. *Scientific reports* 11, 1 (2021), 13416.

[8] Lucio La Cava, Luca Maria Aiello, and Andrea Tagarelli. 2023. Drivers of social influence in the Twitter migration to Mastodon. *Scientific Reports* 13, 1 (2023), 21626.

[9] Michele Coscia. 2021. The atlas for the aspiring network scientist. *arXiv preprint arXiv:2101.00863* (2021).

[10] Michele Coscia and Frank MH Neffke. 2017. Network backboning with noisy data. In *2017 IEEE 33rd international conference on data engineering (ICDE)*. IEEE, 425–436.

[11] Samuel P Fraiberger, Roberta Sinatra, Magnus Resch, Christoph Riedl, and Albert-László Barabási. 2018. Quantifying reputation and success in art. *Science* 362, 6416 (2018), 825–829.

[12] Anna Gallo, Fabio Saracco, and Tiziano Squartini. 2025. Statistically validated projection of bipartite signed networks. *npj Complexity* 2, 1 (2025), 22.

[13] Valerio Gemmetto, Alessio Cardillo, and Diego Garlaschelli. 2017. Irreducible network backbones: unbiased graph filtering via maximum entropy. *arXiv preprint arXiv:1706.00230* (2017).

[14] Sandra González-Bailón, David Lazer, Pablo Barberá, Meiqing Zhang, Hunt Allcott, Taylor Brown, Adriana Crespo-Tenorio, Deen Freelon, Matthew Gentzkow, Andrew M Guess, et al. 2023. Asymmetric ideological segregation in exposure to political news on Facebook. *Science* 381, 6656 (2023), 392–398.

[15] Daniel Grady, Christian Thiemann, and Dirk Brockmann. 2012. Robust classification of salient links in complex networks. *Nature communications* 3, 1 (2012), 864.

[16] Furkan Gursoy and Bertan Badur. 2021. Extracting the signed backbone of intrinsically dense weighted networks. *Journal of Complex Networks* 9, 5 (2021), cnab019.

[17] Qiuyi Han, Kevin Xu, and Edoardo Airoldi. 2015. Consistent estimation of dynamic and multi-layer block models. In *International Conference on Machine Learning*. PMLR, 1511–1520.

[18] Sanaa Hmaida, Hocine Cherifi, and Mohammed El Hassouni. 2024. A multilevel backbone extraction framework. *Applied Network Science* 9, 1 (2024), 41.

[19] Woo Seong Jo, Jaehyuk Park, Arthur Luhur, Beom Jun Kim, and Yong-Yeol Ahn. 2020. Extracting hierarchical backbones from bipartite networks. *arXiv preprint arXiv:2002.07239* (2020).

[20] Alec Kirkley. 2025. Fast nonparametric inference of network backbones for weighted graph sparsification. *Physical Review X* 15, 3 (2025), 031013.

[21] Alec Kirkley, George T Cantwell, and Mark EJ Newman. 2019. Balance in signed networks. *Physical Review E* 99, 1 (2019), 012320.

[22] Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P Gleeson, Yamir Moreno, and Mason A Porter. 2014. Multilayer networks. *Journal of complex networks* 2, 3 (2014), 203–271.

[23] Lea Krivaa and Michele Coscia. 2024. Traces Of Unequal Entry Requirement For Illustrious People On Wikipedia Based On Their Gender. *Advances in Complex Systems* 27, 03 (2024), 2450003.

[24] Ajay Kumar, Shashank Sheshar Singh, Kuldeep Singh, and Bhaskar Biswas. 2020. Link prediction techniques, applications, and performance: A survey. *Physica A: Statistical Mechanics and its Applications* 553 (2020), 124289.

[25] Morgane Laouenan, Palaash Bhargava, Jean-Benoît Eyméoud, Olivier Gergaud, Guillaume Plique, and Etienne Wasmer. 2022. A cross-verified database of notable people, 3500BC-2018AD. *Scientific Data* 9, 1 (2022), 290.

[26] Jean-Marie Lehn. 2013. Perspectives in chemistry—steps towards complex matter. *Angewandte Chemie International Edition* 52, 10 (2013), 2836–2850.

[27] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Predicting positive and negative links in online social networks. In *Proceedings of the 19th international conference on World wide web*. 641–650.

[28] 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*. 1361–1370.

[29] Daqing Li, Bowen Fu, Yunpeng Wang, Guangquan Lu, Yehiel Berezin, H Eugene Stanley, and Shlomo Havlin. 2015. Percolation transition in dynamical traffic network with evolving critical bottlenecks. *Proceedings of the National Academy of Sciences* 112, 3 (2015), 669–672.

[30] Ruikun Li, Huandong Wang, Jinghua Piao, Qingmin Liao, and Yong Li. 2024. Predicting long-term dynamics of complex networks via identifying skeleton in hyperbolic space. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1655–1666.

[31] Federico Malizia, Alessandra Corso, Lucia Valentina Gambuzza, Giovanni Russo, Vito Latora, and Mattia Frasca. 2024. Reconstructing higher-order interactions in coupled dynamical systems. *Nature Communications* 15, 1 (2024), 5184.

[32] Riccardo Marcaccioli and Giacomo Livan. 2019. A pólya urn approach to information filtering in complex networks. *Nature communications* 10, 1 (2019), 745.

[33] Víctor Martínez, Fernando Berzal, and Juan-Carlos Cubero. 2016. A survey of link prediction in complex networks. *ACM computing surveys (CSUR)* 49, 4 (2016), 1–33.

[34] Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. 2011. DBpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*. 1–8.

[35] Federico Musciotto, Federico Battiston, and Rosario N Mantegna. 2021. Detecting informative higher-order interactions in statistically validated hypergraphs. *Communications Physics* 4, 1 (2021), 218.

[36] Zachary Neal. 2014. The backbone of bipartite projections: Inferring relationships from co-authorship, co-sponsorship, co-attendance and other co-behaviors. *Social Networks* 39 (2014), 84–97.

[37] Zachary P Neal. 2022. backbone: An R package to extract network backbones. *PloS one* 17, 5 (2022), e0269137.

[38] Zachary P Neal, Rachel Domagalski, and Bruce Sagan. 2021. Comparing alternatives to the fixed degree sequence model for extracting the backbone of bipartite projections. *Scientific reports* 11, 1 (2021), 23929.

[39] Mark EJ Newman. 2018. Network structure from rich but noisy data. *Nature Physics* 14, 6 (2018), 542–545.

[40] RL Ott and Michael Longnecker. 2010. *An introduction to statistical methods and data analysis*. Cengage Learning Inc.

[41] A Roxana Pamfil, Sam D Howison, Renaud Lambiotte, and Mason A Porter. 2019. Relating modularity maximization and stochastic block models in multilayer networks. *SIAM Journal on Mathematics of Data Science* 1, 4 (2019), 667–698.

[42] Tiago P Peixoto. 2018. Reconstructing networks with unknown and heterogeneous errors. *Physical Review X* 8, 4 (2018), 041011.

[43] Filippo Radicchi, José J Ramasco, and Santo Fortunato. 2011. Information filtering in complex weighted networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 83, 4 (2011), 046101.

[44] Stephany Rajeh, Marinette Savonnet, Eric Leclercq, and Hocine Cherifi. 2022. Modularity-based backbone extraction in weighted complex networks. In *International conference on network science*. Springer, 67–79.

[45] Lucía S Ramírez, Roya Aliakbarisani, M Serrano, and Marián Boguñá. 2025. Extracting the geometric backbone of bipartite networks. *arXiv preprint arXiv:2509.18726* (2025).

[46] M Ángeles Serrano, Marián Boguná, and Alessandro Vespignani. 2009. Extracting the multiscale backbone of complex weighted networks. *Proceedings of the national academy of sciences* 106, 16 (2009), 6483–6488.

[47] Paul B Slater. 2009. A two-stage algorithm for extracting the multiscale backbone of complex weighted networks. *Proceedings of the National Academy of Sciences* 106, 26 (2009), E66–E66.

[48] Natalie Stanley, Saray Shai, Dane Taylor, and Peter J Mucha. 2016. Clustering network layers with the strata multilayer stochastic block model. *IEEE transactions on network science and engineering* 3, 2 (2016), 95–105.

[49] Lovro Šubelj. 2018. Convex skeletons of complex networks. *Journal of The Royal Society Interface* 15, 145 (2018).

[50] Ye Tian, Daniel S Margulies, Michael Breakspear, and Andrew Zalesky. 2020. Topographic organization of the human subcortex unveiled with functional connectivity gradients. *Nature neuroscience* 23, 11 (2020), 1421–1432.

[51] Toni Valles-Catala, Francesco A Massucci, Roger Guimera, and Marta Sales-Pardo. 2016. Multilayer stochastic block models reveal the multilayer structure of complex networks. *Physical Review X* 6, 1 (2016), 011036.

[52] Dan J Wang, Xiaolin Shi, Daniel A McFarland, and Jure Leskovec. 2012. Measurement error in network data: A re-classification. *Social Networks* 34, 4 (2012), 396–409.

[53] Zixiang Wang, Jian Yang, Tongliang Li, Linzheng Chai, Jiaheng Liu, Ying Mo, Jiaqi Bai, and Zhoujun Li. 2023. Multilingual entity and relation extraction from unified to language-specific training. In *Proceedings of the 2023 International Conference on Electronics, Computers and Communication Technology*. 98–105.

[54] Ali Yassin, Hocine Cherifi, Hamida Seba, and Olivier Togni. 2025. Backbone extraction through statistical edge filtering: A comparative study. *PloS one* 20, 1 (2025), e0316141.

[55] Ali Yassin, Hocine Cherifi, Hamida Seba, and Olivier Togni. 2025. Exploring weighted network backbone extraction: A comparative analysis of structural techniques. *PloS one* 20, 5 (2025), e0322298.

[56] Amy Zhao Yu, Shahar Ronen, Kevin Hu, Tiffany Lu, and César A Hidalgo. 2016. Pantheon 1.0, a manually verified dataset of globally famous biographies. *Scientific data* 3, 1 (2016), 150075.

[57] Muhan Zhang and Yixin Chen. 2018. Link prediction based on graph neural networks. *Advances in neural information processing systems* 31 (2018).

[58] Ronda J Zhang, H Eugene Stanley, and Fred Y Ye. 2018. Extracting h-backbone as a core structure in weighted networks. *Scientific reports* 8, 1 (2018), 14356.

[59] XueZhong Zhou, Jörg Menche, Albert-László Barabási, and Amitabh Sharma. 2014. Human symptoms–disease network. *Nature communications* 5, 1 (2014), 4212.