CS224W Final Report:
An Efficient Algorithm for Local Higher-Order Community Detection

Joan Creus-Costa Matthew Das Sarma

December 10, 2017

1 Abstract

We design a local higher-order community detection algorithm based on motif conductance that outperforms SNAP by a factor of about 60 in the precomputation of the non-local helper matrices. Working with the Wikipedia graph and considering categories on Wikipedia as ground-truth communities, we identify a three node motif with respect to which local higher-order clustering achieves optimal clustering quality relative a Jaccard similarity based metric similar to the well-established purity metric. The identified motif, M_3 (two undirected edges and one directed edge), significantly outperforms the induced undirected triangle for local higher-order clustering across a wide range of hyper-parameters on the Catalan Wikipedia graph. We also analyze how the quality and performance of clustering responds to changes in various parameters and observe that each motif has different importance across the various editions of Wikipedia.

2 Introduction

Popular notions of graph clustering and partitioning often require nodes to belong to at most one cluster. However, many forms of structured data are naturally categorized into overlapping clusters. For example, categories on Wikipedia reflect a semantic relationship between pages that is not mutually exclusive. Additionally, categories on Wikipedia are often hierarchical, with large categories containing many subcategories of greater specificity. In this project, we seek to use recent ideas in higher-order local graph clustering [1, 2] to develop an algorithmic approach for grouping nodes into possibly overlapping clusters that approximate natural multi-clustering relationships. In particular, given multiple graphs formed from the same generative process, we hope to use the multi-clustering labels on some graphs to predict the multi-clusterings over the other graphs. In the subsequent sections, we shall define our cluster quality metric, clustering algorithms, supervised learning model, and the optimizations and approximations required to achieve tractability on large graphs. We note that Wikipedia is a particularly interesting public dataset for this research because Wikipedia is offered in a multitude of languages, each of which induces a page-link graph generated from a similar underlying process.

2.1 Review of literature

Many of the fastest clustering algorithms rely on spectral properties of graphs to quickly identify low conductance cuts. Cheeger's inequality provides a relationship between the lowest conductance cut in a graph and the second smallest eigenvalue of its Laplacian [3]. *Local Graph Partitioning Using PageRank Vectors* [4] proposes a local approach, which identifies low conductance cuts in a graph in time proportional to the size of the cut by simulating random walks from a seed node.

The algorithm of Andersen et al. [4] treats the "edge" as the basic unit of connectivity. However, an emerging area of research explores higher-order units of connectivity such as small "motif" subgraph, including triangles and claws. Higher-order graph clustering frameworks have especially been discussed in the context of triangle connectivity and triangle-

based clustering [1, 2, 5, 6]. The 2017 paper *Local Higher-Order Graph Clustering* [2] proposes *motif conductance*, which generalized edge conductance by capturing the probability that a random endpoint of a random motif adjacent to a random node from the cluster is outside of the cluster, as an alternative clustering metric. They adapt the method championed by Andersen et al. [4] by reducing the problem of motif clustering over a directed graph to local graph clustering over a weighted, directed graph [1, 2].

Benson et al. [1] suggests a new class of semi-local higher-order clustering methods that first precompute a global $|V| \times |V|$ matrix W_M that serves as a weighting factor, which transforms the instance of M-motif clustering into an undirected local conductance-based clustering problem. W_M may be expensive to compute because it requires the computation across all pairs $(u, v) \in V$ of the number of instances of M that contain (u, v). Building off of Benson et al. [1], Yin et al. [2] conducts a variety of experiments based on motif conductance, particularly with triangle motifs. Their paper briefly discusses an application of three-node motif clustering to Wikipedia's page-link graph, using 100 Wikipedia categories as the ground truth for validation purposes. While the paper suggests that edge-conductance based clustering achieves superior F_1 score and recall and only marginally worse precision than motif based clustering, the paper does not elaborate on their methodology or discuss the results in any detail.

2.2 Definitions

Given a collection of elements U and a multiset $\mathcal{U}$ of subsets of U, we define the *multi-clustering* of U induced by $\mathcal{U}$ to be the directed graph defined over the vertex set of $\mathcal{U} \cup \{\{u\} \mid u \in U\}$ with edges defined by the "subset of" relation.

A Wikipedia is comprised of a *page-link graph*, denoted $G_p = (V_p, E_p)$, which has pages as vertices and links between pages induce directed multi-edges, and a *category graph*, denoted $G_c = (V_c, E_c)$, which has pages and categories as vertices and the "contains" relation induces directed edges. Notice that the category graph of a Wikipedia may be viewed as a multi-clustering of the corresponding page-link graph.

3 Dataset

Each of Wikipedia's almost three hundred language editions has a publicly available, regularly updated dump of all of their contents at [7]. Dumps include both raw contents as well as database tables concerning its structure, e.g. page links and category membership relations, which respectively define two graphs. This provides a very rich dataset, with hundreds of instances that come from a similar generative process that can be used to evaluate algorithms trained on graphs that span orders of magnitude in size. This makes it possible to perform quick experiments on smaller language editions until definitive results can be obtained on larger, more established editions. We arbitrarily chose to work with the Scots ($|V| = 5 \times 10^4$, $|E| = 1.3 \times 10^6$), Catalan ($|V| = 6 \times 10^5$, $|E| = 4 \times 10^7$) and English ($|V| = 6 \times 10^6$, $|E| = 4 \times 10^8$) editions.

3.1 Data collection

Database dump collection from [7] is automated, fetching the latest dumps for page links, category membership, redirections and page metadata. The raw dumps require significant cleanup: redirections between pages have to be followed (e.g. links to "United States of America" should be treated as links to the canonical name "United States'), links to missing pages are ignored, and various inconsistencies from automated dumps are identified. Each page $p \in V_p$ and category $c \in V_c$ is given a unique increasing numeric identifier, and the edges between pages and categories are stored in a compact binary representation. To allow for easier debugging, those can be referenced to the human-readable page name corresponding to a particular identifier.

A pruning step identifies categories that contain to what Wikipedia refers to as "tracking" categories that have no semantic or encyclopedic value and are therefore orthogonal to the clustering problem. Those include categories such as "Pages with coordinates" or "Stubs."

3.2 Dataset characterization

We performed an initial characterization of the dataset with the help of SNAP [8] as well as intuitive metrics to evaluate the tractability of the problem. This analysis was mostly performed on the Catalan language edition of Wikipedia, with little expected loss of generality with respect to the English edition. Intuitively, past some threshold size, the encyclopedic core of a Wikipedia will remain similar with a logarithmic-like growth. Indeed, Figure 1 compares some graph metrics across different language editions, that share the

same general features; extrapolating, similar results would be expected from the English edition. Furthermore, the editorial standards (which effectively affect the generative model from where G_p and G_c come from) are fairly similar.

We noted the presence of a very large SCC, with 528,431 of the 559,458 nodes. Another feature apparent in the data is the somewhat artificial presence of cliques due to the presence of templates that link to related pages at the end of pages. Figure 2 shows the in- and out-degree distribution for the page-link graph: outliers in the distribution can be explained by the presence of those cliques. Otherwise, the in-degree distribution looks like a power law, up to some cutoff, and the out-degree distribution is a patched function with a maximum at about 10 links into the page.

We evaluated a non-standard metric to get a preliminary idea of how feasible clustering is: one concern is that, if the category structure of Wikipedia is not sufficiently orthogonal (i.e. the same page has multiple categories that overlap) it becomes hard to tell them apart and therefore perform successful clustering. While that is sometimes the case, we randomly sampled pairs of categories from V_c and computed the Jaccard similarity between the pages contained in each category:

$$\rho = \frac{|p_1 \cap p_2|}{|p_1 \cup p_2|} \quad p_i = \{p \mid (p, c_i) \in E_c\} \text{ for } c_1, c_2 \in V_c$$

Figure 3 shows the distribution of this metric over a random sample of $N = 5000$ pairs of categories for the Catalan edition of Wikipedia. We can see that 80% of the randomly sampled pairs had a similarity below 10%—this seems to indicate that most categories can be "told apart" and, while not perfectly orthogonal, the category structure is reasonable. Here the effect of pruning meta-categories was very noticeable, given that internal Wikipedia categories tend to have a lot of overlap between unrelated pages (e.g. people born on a given year, or pages missing an image).

4 Methodology

4.1 GPU accelerated local high-order clustering

In "Local High-Order Clustering" [2], the implementation of the algorithm requires the computation of W_{ij}, a symmetric, sparse matrix that for a given motif M counts the number of times the pair of nodes $(i, j) \in (V \times V)$ participates in the given motif. This is a costly operation: it is the only non-local step in the otherwise local algorithm and requires enumerating motifs (which is particularly hard for high-order motifs); the size of the matrix is quadratic in the number of nodes, and for each node we'll have to inspect a number of edges quadratic in the degree of the node.

This fact might be acceptable if motif clustering is being executed on a single dataset, given that it can be precomputed and thereafter used inexpensively. However, our work involves comparing the usefulness of different motifs on multiple datasets, which makes this computation a non-constant cost. The need to be able to handle large datasets, such as the English-language Wikipedia (with five million nodes and more than four hundred million edges), also constrains the required performance.

In order to be able to efficiently compute this matrix we implemented a parallel version of the motif counting performed in [1]. As in the original paper, we define an ordering of the nodes, sorting them by their total degree. For each node u, we select pairs of neighbors (v, w) that come higher in the ordering, and for each pair (which defines a triangle, along with u) we evaluate the presence of a motif. This evaluation step is mapped onto the GPU, where it is executed in parallel. A reduction step combines the outputs onto a sparse matrix, stored as a hash map.

4.2 Multi-clustering similarity

Given a universe U and two hierarchical multi-clusterings $\mathcal{A}$ and $\mathcal{B}$, we define the similarity of $\mathcal{A}$ onto $\mathcal{B}$ based on the Jaccard similarity of the optimal (non-exclusive) matching:

$$\mathrm{Sim}'(\mathcal{A}, \mathcal{B}) = \frac{1}{|\mathcal{A}|} \sum_{A \in \mathcal{A}} \max_{B \in \mathcal{B}} \mathrm{Jac}(A, B)$$

A more complete similarity between $\mathcal{A}$ and $\mathcal{B}$ is given by $\mathrm{Sim}(\mathcal{A}, \mathcal{B}) = \mathrm{Sim}'(\mathcal{A}, \mathcal{B}) + \mathrm{Sim}'(\mathcal{B}, \mathcal{A})$, which accounts for the similarity of each clustering onto the other. Defining the k-minhash $h_k(S)$ of set S as the set of up to k elements $s \in S$ that minimize $h(s)$ for some hash function h. It is well known for $A, B \subset U$ that

$$\widehat{\mathrm{Jac}(A, B)} = \frac{h_k(h_k(A) \cup h_k(B)) \cap h_k(A) \cap h_k(B)}{k}$$

is an unbiased estimator of $\mathrm{Jac}(A, B)$ with expected error $O(\frac{1}{\sqrt{k}})$. Sampling n i.i.d. sets from $\mathcal{A}$ as $\mathcal{A}'$, we have the estimator

$$\widehat{\mathrm{Sim}'(\mathcal{A}, \mathcal{B})} = \frac{n}{|\mathcal{A}|} \sum_{A \in \mathcal{A}'} \max_{B \in \mathcal{B}} \widehat{\mathrm{Jac}(A, B)}$$

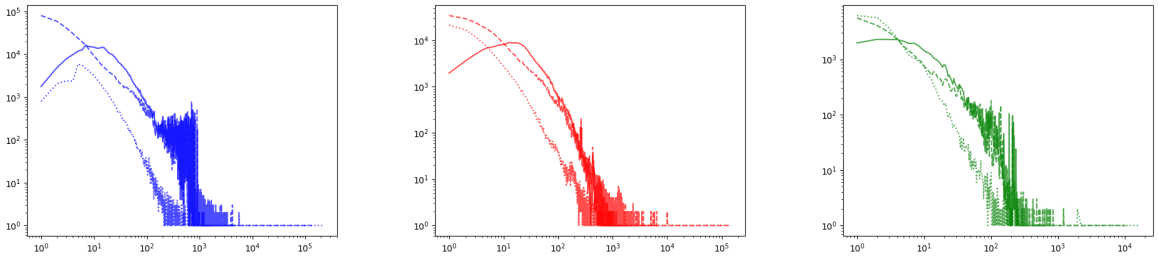
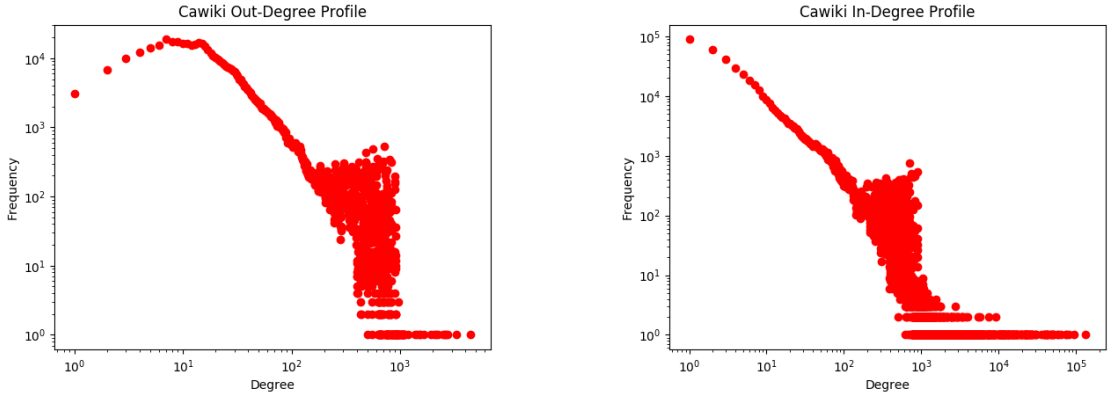


Figure 1: In- (solid), out- (dashed), and category (dotted) degree distributions for the Catalan, Czech and Scots editions of Wikipedia, respectively. We can see that similar features are shared across languages.



(a) Out-degree distribution

(b) In-degree distribution

Figure 2: Out-degree and in-degree distributions of pages on Catalan Wikipedia (cawiki).

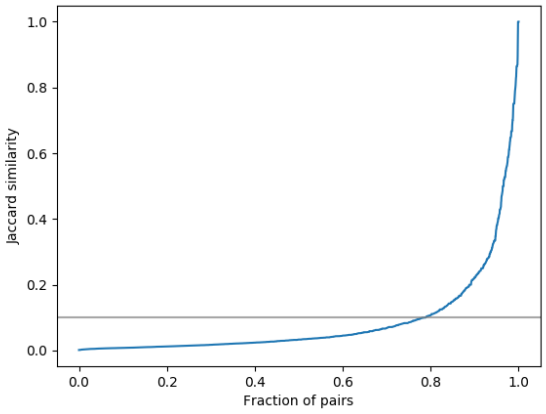


Figure 3: Distribution of Jaccard similarity of randomly sampled pairs of categories for the Catalan language edition of Wikipedia.

$\widehat{\text{Sim}'}(\mathcal{A}, \mathcal{B})$ has expected error $O(\frac{1}{\sqrt{n}} + \frac{1}{\sqrt{k}})$ for a single run. While $\widehat{\text{Sim}'}(\mathcal{A}, \mathcal{B})$ may be computed without an entire multi-clustering defined over $\mathcal{A}$, it requires the k-minhashes to be enumerated over the entire multi-clustering $\mathcal{B}$.

For our problem, we may take q independent hash functions and construct k-minhashes with respect to each of the q hash functions corresponding to the categories $\mathcal{C}$ represented by G_c. These k-minhashes may be precomputed in $O(kq|E_c|)$ time by first topologically sorting G_c and then iteratively constructing the k-minhashes. We use Tarjan's algorithm to simultaneously eliminate cycles in $\mathcal{C}$ (which are semantically meaningless by the inclusion relation) and provide a topological ordering over V_c.

Then, given a set of n clusters $\mathcal{C}'$ generated by some clustering procedure over G_p, we may compute $\widehat{\text{Sim}'}(\mathcal{C}', \mathcal{C})$ in $O(nkq|C|)$ time with expected error $O(\frac{1}{\sqrt{nq}} + \frac{1}{\sqrt{kq}})$. While it would be desirable to also compute $\widehat{\text{Sim}'}(\mathcal{C}, \mathcal{C}')$, it is generally infeasible during simulation to compute a full multi-clustering $\mathcal{C}'$ over G_p. Since the variance of the maximum function is unbounded, we cannot approximate $\widehat{\text{Sim}'}(\mathcal{C}, \mathcal{C}')$ by only considering n clusters sampled from $\mathcal{C}'$ as we did in the forwards direction in general.

4.3 Motif conductance combination

Given k motifs $M_1, \cdots, M_k$, we may precompute the matrices $W_{M_1}, \cdots, W_{M_k}$. We define $C(\beta_1, \cdots, \beta_k)$ to be the clustering obtained by our algorithm taking $W = \beta_1 W_{M_1} + \cdots + \beta_k W_{M_k}$ as our normalization matrix, conditioned on some hyperparameters that shall be described later. For a given set of hyperparameters, then, we may phrase our learning problem as the optimization problem:

$$\vec{\beta}^* = \arg\max_{\|\vec{\beta}\|=1} \text{Sim}(C(\vec{\beta}), \mathcal{C})$$

We plan to approximate $\vec{\beta}^*$ through a variety of optimization methods. To make this computationally feasible, we will either need to approximate $C(\vec{\beta})$ as a small constant number of clusters (which, formally, removes all statistical guaranties on $\text{Sim}(C(\vec{\beta}), \mathcal{C})$), or we need to relax our similarity metric to be the L_2 approximation of the L_∞-norm, which shall allow gradient descent.

The clustering algorithm of Andersen et al. [4] requires parameters α and ϵ, which influence the drift and potential cut of the clusters. Since the distribution of cluster sizes spans several orders of magnitude, we shall fit hyperparameter λ and draw ϵ from the exponential distribution with parameter λ. We assumed that α was constant at 0.95.

One problem is the difficulty of training on a particular dataset without overfitting; partitioning or subsampling the graph could bias the clustering algorithm and provide an incomplete picture of the results. In order to show that this algorithm is able to generalize well, and avoid overfitting in the results, we want to attempt to perform training on different language editions to Wikipedia and test on separate editions. Thus it is possible to claim that such an algorithm would be able to learn from graphs sampled from a somewhat similar generative distribution (in this case, that of page-links graphs of free encyclopedias).

	Wiki	**Motif**	**Runtime**
SNAP	ca	M_1	18
		M_2	64
		M_3	71
		M_4	30
		M_5	20
		M_6	22
		M_7	20
This work	ca	$M_1 - M_7$	4
	en	$M_1 - M_7$	45

Table 1: Comparison of the runtime (in minutes) of the computation of the W matrices for each motif M_i, as executed on an off-the-shelf laptop with an Intel i5-5200U and corresponding integrated Intel graphics card. Note that this work computes all motif matrices simultaneously and was able to compute motif matrices for the English Wikipedia (the SNAP function was not able to compute it after a few hours).

5 Results

5.1 Hardware accelerated motif adjacency

We implemented hardware accelerated motif adjacency computation as described above by using the OpenCL standard on a C++ interface. The entire graph is stored in memory (at approximately 2.8 GB, even the English Wikipedia graph fits on a modern GPU); however, for bigger graph it could easily be extended to remove this requirement. It is stored as

a list of sorted lists of neighbors, where the first two bits indicate the direction of the edge. As such, there is a factor of 2 spatial redundancy, that helps with performance (the existence and character of edges, the critical step that the GPU computes, performs binary searches over those lists). While we implemented some of the wedge motifs (that do not involve edges between two of the nodes), those make the matrix sparsity quickly explode and are thought to not have a big impact on clustering (see later discussion).

We observed a significant speedup over a single-threaded CPU implementation of the same algorithm as publicly open sourced by [1]. Excluding graph loading times (that are twice as short in this work due to less overhead) we found that our code was about 59 times faster at computing all the motifs in the Catalan Wikipedia (see Table 5.1). In addition, we were able to compute the W matrices for $M_1 - M_7$ on the English-language edition of Wikipedia in just 45 minutes—shorter than it takes a CPU implementation to compute some single motifs on the much smaller Catalan edition.

5.2 Simplified Jaccard Similarity

Our original intent was to define the Jaccard similarity in terms of the symmetric definition $\text{Sim}(\mathcal{C}, \mathcal{C}') = \text{Sim}'(\mathcal{C}, \mathcal{C}') + \text{Sim}'(\mathcal{C}', \mathcal{C})$. However, this approach suffers from computational intractability since it is generally infeasible to fully compute a sufficiently large ($|\mathcal{C}| = 7 \times 10^4$ for Catalan Wikipedia) local clustering to compute the optimal matching onto the ground-truth categories. Additionally, Figure 4(a) shows that for Catalan Wikipedia and a generic adjacency matrix $W = \frac{1}{8}W_{M_1} + ... + \frac{1}{8}W_{M_7}$ the clustering quality $\text{Sim}(\mathcal{C}, \mathcal{C}')$ increases monotonically with $|\mathcal{C}'|$. This implies that the similarity metric, in this particular problem, encourages clusterings far larger than the number of communities in the ground-truth $\mathcal{C}$.

Instead, we consider the simplified Jaccard similarity of a (partial) clustering $\mathcal{C}'$ onto the ground-truth categories $\mathcal{C}$. This operations is longer symmetric, but only considers the quality of a matching of one clustering onto another clustering. When $\mathcal{C}'$ is n clusters sampled i.i.d. from a larger family of clusters (e.g. $C(\vec{\beta})$), $\text{Sim}'(\mathcal{C}', \mathcal{C})$ approximates $\text{Sim}'(C(\vec{\beta}), \mathcal{C})$ with error $O(1/\sqrt{|\mathcal{C}'|})$. Figure 4(b) shows that $\text{Sim}'(\mathcal{C}', \mathcal{C})$ does not grow monotonically with $|\mathcal{C}'|$, but instead its accuracy as an estimator of the true value improves with sample size (contrast with Figure 4(a)). Clustering methods that perform well under this modified definition of similarity are succeed at predicting local categories on Wikipedia, but are not necessarily good at predicting *representative* categories on Wikipedia, which would be captured by $\text{Sim}'(\mathcal{C}, C(\vec{\beta}))$ instead.

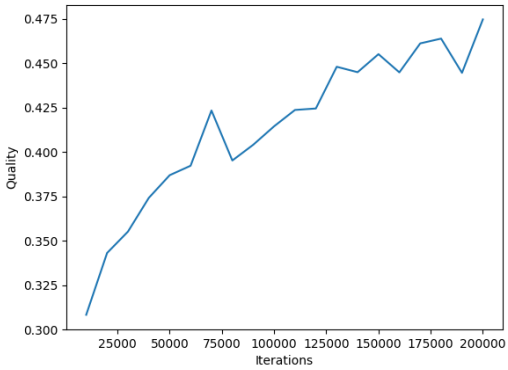
Although our minhash approximation $\widehat{\text{Sim}'(\mathcal{C}', \mathcal{C})}$ is not independent across multiple clusterings $\mathcal{C}'$ for a fixed hash function h, we find that, in practice, it is sufficient to only use only a single hash function. On Aragonese Wikipedia, we found that the standard deviation of $\widehat{\text{Sim}'(\mathcal{C}', \mathcal{C})}$ for $\mathcal{C}'$ sampled from $C(\vec{\beta})$ and multiple hash functions is less than 0.006 across many different $\vec{\beta}$. Since we set the length of each minhash to at most $k = 400$, the expected error is already about five percent. Consequently, we found that empirically any additional precision that could stem from using multiple hash functions would be negligible compared to the variance already inherent in the estimator.

5.3 Motifs on Catalan Wikipedia

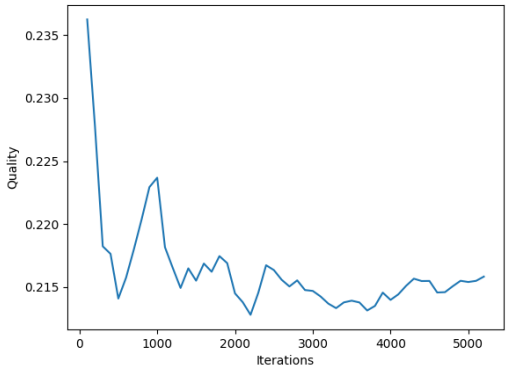
Since we consider motifs as induced subgraphs, n-vertex graph space may be expressed as a vector space over the induced motif adjacency matrices W_M for each three node motif M. We, however, limit ourselves to only the seven motifs M_1 through M_7 given in Figure 5 and we also treat the undirected adjacency matrix M_0 as a motif. This allows our formulation $M = \beta_0 W_{M_0} + ... + \beta_7 W_{M_7}$ to still span the edge space of G while reducing the total number of parameters that need to be fit. A priori, we also believed that the excluded "wedge" motifs, which do not contain at least one edge between every pair of the nodes in the motif, contain less information than the included seven motifs. We leave exploration of fitting a clustering model over all three node motifs to future research.

Unfortunately, $\text{Sim}(C(\vec{\beta}))$ is not linear or else it would be relatively easy to optimize relative to the basis W_{M_0} through W_{M_k}. While $\text{Sim}(C(\vec{\beta}))$ is not linear, $\text{Sim}(C(\vec{\beta}))$ is stable relative small perturbations to $\vec{\beta}$. We do not formally show this fact, but the inverse dependencies on W_{ij} in the motif clustering algorithm seem to imply that a small perturbation to $\vec{\beta}$ can only have marginal impact limited to the frontiers of the clusters in $C(\vec{\beta})$. This permits $\text{Sim}(C(\vec{\beta}))$ to be optimized through the use of probabilistic optimization techniques, such as simulated annealing or differential evolution, which shall be discussed in more detail in Section 5.5.

A good starting point for optimizing the Jaccard similarity based quality of clusterings of G_p parameterized by $\vec{\beta}$ is to study the behavior of $C(\vec{\beta})$ with respect to each basis vector of the motif space. We



(a) $\text{Sim}(\mathcal{C}, C(\vec{\beta}))$ as a function of $|C(\vec{\beta})|$ for $\beta = \frac{1}{8}\mathbf{1}$. This study uses Aragonese Wikipedia ($|V_p| \approx 3 \times 10^4$) such that large clusterings could feasibly be calculated.



(b) $\text{Sim}'(C(\vec{\beta}, \mathcal{C}))$ as a function of $|C(\vec{\beta})|$ for $\beta = \frac{1}{8}\mathbf{1}$. This study uses uses Aragonese Wikipedia for comparability to Figure 4(a).

Figure 4: We relax the similarity metric $\text{Sim}(\mathcal{C}', \mathcal{C})$ to $\text{Sim}'(\mathcal{C}', \mathcal{C})$ such that similarity can be computed with expected error $O(1/\sqrt{|\mathcal{C}'|})$.

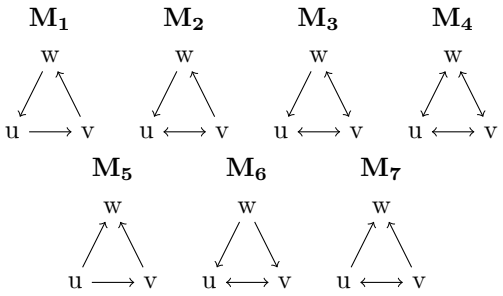


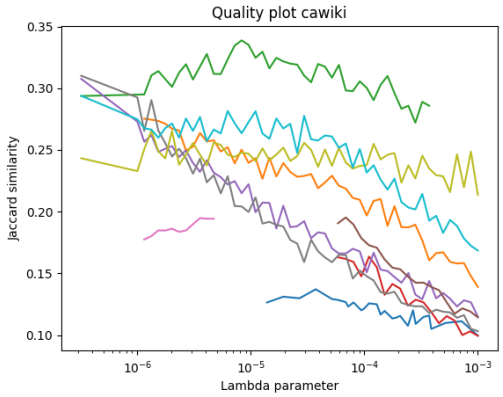
Figure 5: Nomenclature for motifs as used in [1]. Note that this does not include "wedge" motifs that do not involve edges among all three participants.

consider clusterings $C_i = C(W_{M_i})$ over the page-link graph of Catalan Wikipedia. We find remarkably different quality of these clusterings when varying the distributional parameter λ (which we normalize by $\|M_i\|_1$ to remain scale invariant) in Figure 6(a). As λ increases, the expected size of the clusters in C_i decreases, as does the quality of C_i relative to the ground-truth communities. The quality of clustering C_3 exceeds the quality of the other basic clusterings across all values of λ considered except for $\lambda = 10^{-7}$. For very small λ, local clusterings may be very large – often approaching the size of the entire strongly connected component induced by the motif. Therefore, the quality of C_i for small λ is highly dependent on the global structure of G_p as opposed to the local structure. So, it may be reasonable to conclude that in the relevant territory of $\lambda \in [10^{-6}, 10^{-4}]$, the optimal single motif to use to locally cluster Catalan Wikipedia is M_3.

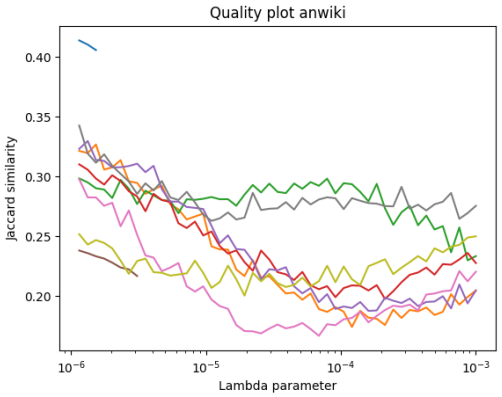
5.4 Clustering Algorithm Performance

The motif-based local clustering procedure is very efficient for large λ, but deteriorates rapidly as λ approaches zero. Figure 7 show the computational time of different clusters as a function of ϵ for a specific motif (M_1 in this case). The expected compute time is approximately quadratic in the inverse of ϵ, which is consistent with the theoretical worst-case guaranties of higher-order local clustering. Since ϵ is sampled from the exponential distribution parameterized by λ, we have that the expected time required to compute $C(\vec{\beta}; \lambda)$ grows with λ.

We threshold the exponential distribution such that ϵ is never less than $0.1\lambda^{-1}$. This threshold is important because the standard deviation of the exponential distribution is λ^{-2}. This threshold eliminates the probability of tail events that would cause the algorithm to run in $O(\lambda^{-4})$ time due to the inverse quadratic time dependence on $\epsilon \sim \text{Exp}(\lambda)$. In practice, the threshold also does not appear to significantly change the distribution of cluster sizes resulting from our algorithm for appropriate values of λ. Figure 8(a) gives an example of the distribution of cluster sizes as a function of ϵ for some motif when ϵ is drawn from a thresholded exponential distribution. While Figure 8(a) is sparser than 8(b) due to a smaller sample size,



(a) From the top-most on the right to the bottom-most: green) M_3; yellow) M_4; teal) M_7; orange) M_0; purple) M_6; brown) random unit vector; gray) M_1; blue) M_5; red) $\frac{1}{8}\mathbf{1}$. Defined between $\lambda = 10^{-6}$ and $\lambda = 10^{-5}$, pink) M_2. Due to computational limitations and the particularly poor conditioning of M_1, we were unable to compute the quality of all basic motif clusterings across all λ considered.



(b) From the top-most on the right to the bottom-most: gray) M_4; yellow) M_7; green) M_3; red) $\frac{1}{8}\mathbf{1}$; pink) M_2; purple) M_6; orange) M_0. Defined for $\lambda < 10^{-5}$, blue) M_1; brown) M_5. Due to computational limitations and the particularly poor conditioning of M_1 and M_5, we were unable to compute the quality of all basic motif clusterings across all λ considered.

Figure 6: Comparison of wide range of λ values across all individual motif clusterings and a few motif combinations on Catalan Wikipedia (left) and Aragonese Wikipedia (right).

it is notable that the thresholding of ϵ does not appear to have a significant distributional impact on the size of the clusters resulting from $C(\vec{\beta})$. We also note that the variance of cluster sizes increases in the territory between $\lambda = 10^{-6}$ and $\lambda = 10^{-5}$, which we speculate is because clusters are neither 'forced' by a small λ parameter to subsume the entire SCC nor restricted to exclusively very small clusters by a large λ.

Since the quality of the clustering is highly dependent on λ, we must be very careful to tune our distributional parameter. So far, the λ parameter in this paper has been expressed as a true lambda parameter normalized by $L = \beta_0\|W_{M_0}\|_1 + \ldots \beta_7\|W_{M_7}\|_1$. This normalization serves to keep the actual lambda used in for higher-order local clustering as in Yin et al. [2] on the correct order of magnitude relative to the entires in the matrix $W = \beta_0 W_{M_0} + \ldots + \beta_7 W_{M_7}$. This is important because $\|W_M\|_1$ varies by multiple orders of magnitude across the eight motifs, which is enough of a discrepancy to push λ outside of the optimal territory if the true lambda is not normalized by L.

5.5 Approaches to optimizing β

Given the computational expense of computing a cluster, especially for some parts of the parameter space—even if the similarity computation was extremely optimized—we focused our general analysis to generally comparing the performance of individual motifs as various parameters in the algorithm change, which gives a good idea for how sensitive this method to its parameters and what motifs are useful to a graph, with the intuition about its structure that this might bring.

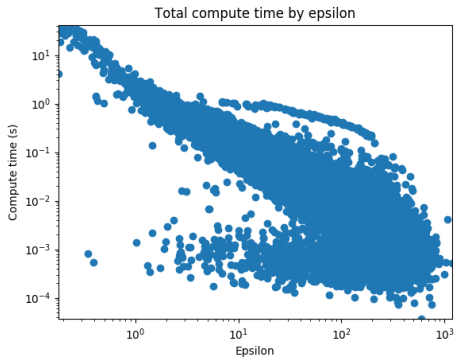
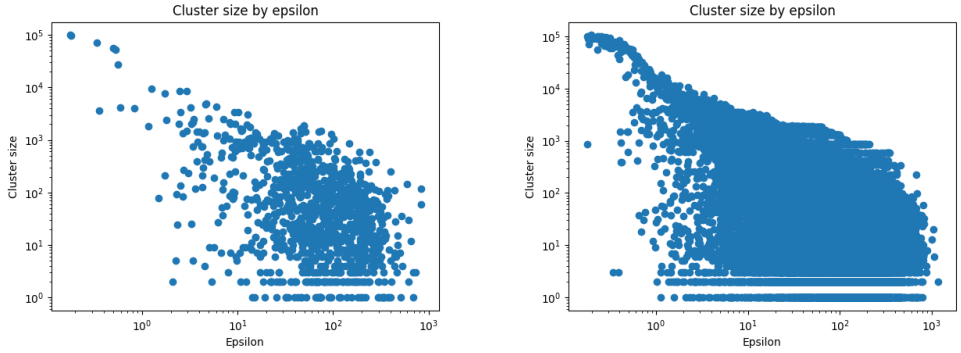


Figure 7: Time taken to compute clusters based on the M_1 motif for different ϵ.

We implemented simulated annealing and randomized coordinate descent, and used an open source implementation of differential evolution to try to maximize the similarity over a thousand trials on a given language edition of Wikipedia. Simulated annealing



(a) Thresholded exponential distribution some motif. (b) Not thresholded exponential distribution M_1.

Figure 8: Effects of thresholding on the exponential distribution.

proved ineffective, randomized coordinate descent was slow, and while differential evolution yielded promising results (in which the convex combination of motifs was better than most motifs individually) more analysis and computation would be required to evaluate its statistical significance.

Somewhat surprisingly, the cost function was not easily optimizable—convex combinations of good motifs did not necessarily give a better result, or yielded a small improvement. One of our experiments involved taking weighing β as the softmax function of the relative performance of each motif (as in Figure 9), which yielded a similarity of 30% (higher than motifs $M_2 - M_7$, but lower than that of M_1 individually). The reasons for this are unclear, but it is hypothesized that the expected decrease in sparsity of the linear combination could worsen the impact of expander subgraphs.

6 Discussion

We believe that motifs and their generalizations can be used to identify communities in graphs that would go undetected without consideration of these higher-order features. In some sense, the transformation from an adjacency matrix A to a motif-adjacency matrix W is a "convolution-like" operation that provides a representation for higher-order features that are difficult to immediately identify in A. While this work develops a scalable system for computing W on a GPU for reasonably large graphs (with millions of nodes, and hundreds of millions of edges), there is a computational bottleneck for large graphs with expander-like components in fitting parameters to combine multiple motif-adjacency matrices into a single matrix. The computational complexity of this optimization is exacerbated by the size of the Wikipedia graph and the distribution of cluster sizes on Wikipedia, which spans multiple orders of magnitude.

We were surprised that although Wikipedias in different languages, intuitively, are approximations of the same semantic relationships between entities, the local higher-order structure of different Wikipedias can be dramatically different. Figure 9 suggests that while local clusterings with respect to M_1 have the least similarity to the ground-truth categories on Catalan and Simple English Wikipedias, it provides by far the strongest signal of M_0 through M_7 on Aragonese Wikipedia (see Figure 6(b)). Unfortunately, trying to leverage strong negative signals for local clustering is computational infeasible since negative edges are impermissible and additively scaling the motif-adjacency matrix would destroy its sparsity.

6.1 Future Work

Viewing the transformation from A to W as a "convolution-like" operation applied to graphs, it would be very interesting to combine the ideas in this paper with neural networks. While we were limited in this study by the computational time to compute the quality of a linear combination of motif-adjacency matrices W_M, there are certainly community detection and local clustering scenarios in which feedback may be much less expensive. For example, if every node in the graph corresponded to exactly one canonical local cluster (or a small number of clusters), then the quality of $\vec{\beta}$ could be approximated with a single call to the clustering routine. A shallow neural network over top of the W_M matrices could also lead to

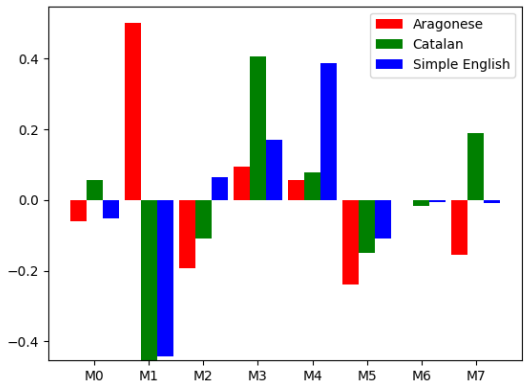


Figure 9: For three sample language editions (Aragonese, Catalan, and Simple English) we computed the similarity metric with all the β mass on each motif separately, and plotted here the relative difference from the mean similarity on each individual motif. Note while some are consistent across language, others (such as M_1, surprisingly) have a big variance. This computation used a fixed $\lambda = 10^{-5}$.

more complicated higher-order feature detection and a better resulting model for community detection.

While our priors suggested that the motifs M_1 through M_7 were more semantically interesting for the problem of local higher-order clustering on Wikipedia, it would be interesting to conduct the same study with respect to all of the three node induced subgraphs of Wikipedia. Future research could also consider other measures of similarity, including convex approximations of our similarity function, which may also enable faster learning of the parameter vector.

Although Wikipedia has a hierarchically structured community graph in which some categories contain other categories, this work disregards the hierarchy by collapsing it into a multi-clustering. It may be possible to construct hierarchical clusterings using higher-order local clustering by varying the parameter ϵ or considering multiple vectors $\vec{\beta}$ for a given node. However, this problem seems even more computational challenging than the task addressed in this work.

7 Contribution

We believe that the authors, Joan Creus-Costa and Matthew Das Sarma, contributed equally to this work.

Matthew Das Sarma characterized the page-link and category graphs of Catalan Wikipedia, developed efficient code for motif-clustering, cluster similarity, and multi-clustering simulation, and profiled the quality of basic motif-based clusterings on Aragonese and Catalan Wikipedia.

Joan Creus-Costa processed and sanitized the Wikipedia datasets, characterized the page-link graphs of multiple Wikipedias, developed GPU accelerated code to compute motif-adjacency matrices over large graphs, and wrote optimization code to achieve optimal higher-order local clustering on Aragonese Wikipedia.

While the Wikipedia dataset is already publicly available, we believe we added value to it by preprocessing so that it is easy to use, without having to deal with the nuances of redirections, missing pages, or malformed dumps. It is available at `https://stanford.edu/~jcreus/cs224w`.

References

[1] Austin R. Benson, David F. Gleich, and Jure Leskovec. Higher-order organization of complex networks. 2016.

[2] Hao Yin, Austin R Benson, Jure Leskovec, and David F Gleich. Local higher-order graph clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 555–564. ACM, 2017.

[3] J Cheeger. A lower bound for the lowest eigenvalue of the laplacian. problems in analysis: A symposium in honor of s. bocher rc gunnind, ed, 1970.

[4] Reid Andersen, Fan Chung, and Kevin Lang. Local graph partitioning using pagerank vectors. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*, pages 475–486. IEEE, 2006.

[5] Christine Klymko, David Gleich, and Tamara G Kolda. Using triangles to improve community detection in directed networks. *arXiv preprint arXiv:1404.5874*, 2014.

[6] Charalampos E Tsourakakis, Jakub Pachocki, and Michael Mitzenmacher. Scalable motif-aware graph clustering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1451–1460. International World Wide Web Conferences Steering Committee, 2017.

[7] Wikimedia downloads. `https://dumps.wikimedia.org`. Accessed: 2017-11-14.

[8] Jure Leskovec and Rok Sosič. Snap: A general-purpose network analysis and graph-mining library. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(1):1, 2016.