Geometric Affinity Propagation for Clustering With Network Knowledge

Omar Maddouri, Xiaoning Qian, *Senior Member, IEEE*, and Byung-Jun Yoon, *Senior Member, IEEE*

***Abstract*—Clustering data into meaningful subsets is a major task in scientific data analysis. To date, various strategies ranging from model-based approaches to data-driven schemes, have been devised for efficient and accurate clustering. One important class of clustering methods that is of a particular interest is the class of exemplar-based approaches. This interest primarily stems from the amount of compressed information encoded in these exemplars that effectively reflect the major characteristics of the corresponding clusters. Affinity propagation (AP) has proven to be a powerful exemplar-based approach that refines the set of optimal exemplars by iterative pairwise message updates. However, a critical limitation is its inability to capitalize on known networked relations between data points often available for various scientific datasets. To address this shortcoming, we propose Geometric-AP, a novel clustering algorithm that effectively extends the original AP to take advantage of the network topology. Geometric-AP obeys network constraints and uses max-sum belief propagation to leverage the available network topology for generating smooth clusters over the network. Extensive performance assessment shows that Geometric-AP leads to a significant quality enhancement of the clustering results when compared to existing schemes. Especially, we demonstrate that Geometric-AP performs extremely well even in cases where the original AP fails drastically.**

***Index Terms*—Affinity propagation, exemplar-based clustering, label smoothing, max-sum belief propagation, message passing, network-based clustering.**

I. Introduction

CLUSTERING refers to the process of partitioning data into groups of points that share specific characteristics where similar instances are assigned to the same cluster. The definition of similarity here is often subjective and greatly depends on the ultimate goal expected from the analysis [1]. This makes clustering a difficult combinatorial problem where researchers continuously strive to develop innovative computational methods to address the increasing complexity associated with new large-scale datasets. The vast majority of current clustering methods are generally fed either with a vector of observations in the feature space or with measures of proximity between data points [2]. Their mission, consequently, is to identify expressive clusters that dissect the dynamics present in the data. Towards this goal, two broad classes of approaches have been proposed. The first set of methods includes models such as kmeans [3] and kmedoids [4] and directly operates on the original feature space to group the data points based on raw pairwise similarities. The other class of approaches maps the manifold structures of the observed feature space into a different latent space where the data might be more separable. Spectral clustering for instance tracks clusters of irregular shapes by leveraging the eigenvalues of the similarity matrix to embed the data into a lower dimensional space [5]. A sub-class of such methods that is of particular interest is the multi-view subspace clustering (MVSC). MSVC aims at leveraging different perspectives of the data for more efficient clustering, where each set of features is considered as a separate view of the observed data. The resulting multiple views are then combined to generate a latent representation in a common subspace that aggregates the information collected from the individual views. However, several limitations have been identified for this class of methods. First, the increasing computational complexity associated with the aggregation of multi-view representations limits the scalability of the clustering method. Another limitation is that there is no guarantee that the generated latent representation will capture the true structure of the common subspace. Furthermore, as each view contributes independently with respect to other views to the optimization of the subspace representation, this makes the algorithm converge to sub-optimal embeddings. To alleviate these issues, various alternatives have been proposed in recent years, where some approaches used linear order complexity learners to extract intermediate view-specific representations in a way to preserve the overall scalability [6]. To better reflect the true subspace structure within the latent representations, an additional latent representation assumption was considered in [7] to better integrate the complete information from the various training views. A recent study [8] investigated the problem of partial multi-view clustering where a joint representation learning and clustering framework has been proposed to enable the extraction of view-specific information during the clustering process based on partial similarity matrices. Nevertheless, to the best of our knowledge, there is currently no method that can simultaneously address all the aforementioned limitations

Manuscript received 10 November 2021; revised 28 November 2022; accepted 5 January 2023. Date of publication 17 January 2023; date of current version 6 October 2023. The work of Xiaoning Qian was supported in part by the National Science Foundation (NSF) under Grants CCF-1553281, IIS-1812641, and IIS-2212419. Recommended for acceptance by Y. Xia. *(Corresponding author: Omar Maddouri.)*

Omar Maddouri is with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: omar.maddouri@tamu.edu).

Xiaoning Qian and Byung-Jun Yoon are with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA, and also with the Computational Science Initiative, Brookhaven National Laboratory, Upton, NY 11973 USA (e-mail: xqian@ece.tamu.edu; bjyoon@ece.tamu.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TKDE.2023.3237630, provided by the authors.

Digital Object Identifier 10.1109/TKDE.2023.3237630

of MVSC. More recent approaches employ deep neural architectures to train non-linear embeddings that better capture the hidden interactions underlying complex systems [9], [10], [11].

Despite the enhanced performance achieved by representation-learning-based approaches, methods that directly operate on the primitive data remain highly sought after in the research community. Exemplar-based approaches present an epitome of such methods that have been extensively applied in diverse fields. For example, exemplars have been widely used in management sciences to find optimal facility locations [12]. In the multi-controller placement problem, exemplars were utilized to locate the best controller sites for software-defined networks to minimize the propagation latency with the switches [13]. Exemplars also play a critical role in recent applications such as electronic commerce systems where the accurate identification of key leaders [14] may yield a fast convergence to the optimal cluster configuration. Affinity propagation (AP) [15] is one of the most appealing exemplar-based clustering methods that have been proposed in recent years. AP iteratively refines the set of candidate points that best exemplify the entire dataset by exchanging messages between all pairs of data points until a set of representatives emerges. Many other implicit exemplar-based methods, including kmeans, consider for "virtual" centroids that may not belong to the original set of data points. For example, each exemplar in kmeans clustering is determined by the average features of the corresponding cluster members. The fact that AP explicitly selects exemplars from the original dataset gives it a significant advantage over other implicit methods–especially, in terms of interpretation and utilization, since the identified exemplars seamlessly relate to many real-world applications. While finding the optimal set of exemplars is an NP-hard problem, AP has proven to be very efficient in realistic settings, being capable of rapidly handling thousands of high dimensional instances. This is enabled by an efficient belief propagation scheme that can take advantage of parallel implementation, which makes AP operate with only $\mathcal{O}(N^2)$ messages, where N represents the total number of data points [15]. In addition to computational efficiency, AP has been shown to yield accurate clustering results, which are relatively insensitive to initialization.

However, one notable shortcoming of the original AP is its limited ability to integrate different levels of information to perform clustering, since it solely relies on pairwise affinities between data points for partitioning the dataset. In various applications such as the discovery of communities in social networks [16], finding functional modules in biological networks [17], [18], or optimizing the usage of communication channels in transportation networks [19], the systems are often described using node features as well as network information. An obvious example that reflects the need for considering the available network information is the study of epidemic spreading on signed networks [20] where the negative or positive interactions between data points significantly help to better understand the influence of structural balance on the dynamics of epidemic spreading in social networks. Unfortunately, AP is unable to leverage such network knowledge to enhance the clustering accuracy. Also, AP faces challenges with datasets of irregularly shaped clusters, sparse datasets, and multi-subclass systems.

A. Previous Work

To alleviate the aforementioned limitations, efforts have been made to adapt AP to specific applications, which we briefly review in this section. In the original formulation of AP, hard consistency constraints have been placed on the elected exemplars that do not refer to themselves [15], which led to efficient identification of convex clusters where data points are well represented by their associated exemplars. To extend AP beyond regularly shaped clusters, a soft-constraint AP (SCAP) method has been proposed, in which the hard constraints have been relaxed. Evaluation on clustering microarray data has shown that SCAP is more efficient than AP in analyzing noisy and irregularly organized datasets [21]. For sparse datasets such as sparse graphs, a fast implementation of AP sets the similarity between unconnected nodes to very small values. This confines the exemplars within direct adjacency of the data points, leading to finely fragmented clusters. To mitigate this shattering pattern, a greedy hierarchical AP (GHAP) algorithm has been proposed [22]. GHAP repeatedly clusters the set of exemplars that emerge from the previous iterations and updates the exemplars labels until a satisfactory coarse clustering is obtained [22]. An evolved theoretical approach for hierarchical clustering by affinity propagation, called Hierarchical AP (HAP), adopts an inference algorithm that disseminates information up and down the hierarchy [23]. HAP outperforms GHAP that clusters only one layer at a time. A semi-supervised AP was proposed in [24], which considers clustering when prior knowledge exists for some pairs of data points indicating their similarity (must-link (ML)) or dissimilarity (cannot-link (CL)). Building on [21], a soft instance-level constraint version has been presented for the semi-supervised AP [25]. Furthermore, AP has been utilized to analyze data streaming dynamics. Instead of operating on high-throughput data, Streaming-AP puts in cascade a weighted clustering step to extract subsets from the data and then performs hierarchical clustering followed by an additional weighted clustering procedure [26]. Another attractive advantage of AP is that it automatically identifies the number of clusters in the data, but its downside is the lack of control over the desired size of the identified clusters. To remedy this limitation, AP has been extended to make the cluster size more manageable. For example, various priors such as the Dirichlet process priors have been integrated into the clustering process for this purpose [27]. Notably, hierarchical clustering principles have been widely utilized to extend the original AP. The main reason is that the single-exemplar design of AP becomes inadequate when applied to model multi-subclass systems. In this regard, a more explicit approach called multi-exemplar affinity propagation (MEAP) has been proposed to address the limitations of AP for multi-subclass problems. In MEAP, two types of exemplars are being identified: A set of sub-exemplars are associated with super-exemplars to approximate the subclasses in the category [28]. MEAP has shown consistent performance in handling problems like scene analysis and character recognition.

As for network information, it has been less considered in exemplar-based clustering literature. Fundamentally, it is more difficult to combine pairwise similarity measures obtained from two different observations: node features and network topology. Additionally, a unified criterion for identifying exemplars and cluster membership based on a compound affinity needs to be determined. For AP, an early attempt employed diffusion kernel similarities obtained using the Laplacian matrix of the network to perform a community detection task [29]. A more recent approach has addressed the same problem by adaptively updating the similarity matrix during the message updates using the degree centrality of potential exemplars [30]. Although tailored similarity measures can slightly improve the efficiency of AP as discussed in [15], they are known to be insufficient and very limited in handling problems with complex underlying structures [27].

B. Extension of Affinity Propagation to Geometric-AP

Motivated by the increasing availability of network information in many structured datasets, this article extends the feature-based affinity propagation (AP) algorithm to a geometric model, which we call Geometric Affinity Propagation (Geometric-AP), where the original energy function is being minimized under additional topological constraints. Indeed, our work builds on top of the latest advances in graph clustering research and endorses two universally accepted properties of connectivity and density for any desired graph cluster. That being said, a good graph cluster should intuitively be connected. Also, its internal density should be significantly higher than the density of the full graph [31]. In the context of our work, we adopt a more lenient definition of connectivity and density as we are not strictly performing graph clustering. Instead, we require that members of each desired cluster should lie within the same region in the network. Additionally, we promote higher internal density of identified clusters by assuming that highly interacting nodes should belong to the same cluster.

To implement the above requirements for AP, we jointly modify the exemplar identification mechanism and the membership assignment procedure to incorporate the connectivity and density properties, respectively. First, we require that a potential exemplar should lie within the local neighborhood of referring nodes with respect to the network. Second, highly interacting nodes must share the same cluster membership. The first connectivity constraint is ensured through an additional penalty term in the optimized net similarity of AP. The second density requirement is secured using a new assignment policy that promotes membership selection among neighbor exemplars. Afterwards, a label smoothing operation is applied to reduce the misassignments and enable a better generalization.

Compared to the original feature-based AP, Geometric-AP has the following advantages.

- It can seamlessly integrate the network information into the clustering setup and notably improve the performance without increasing the model complexity.
- Unlike other approaches, Geometric-AP does not use the network information to tailor the feature-based similarity but instead it jointly employs the node features along with the network information to conduct efficient clustering.

Our fundamental research contribution in this article is to suggest a novel iterative clustering algorithm to minimize an energy-based function that depends on two different measures of similarity as illustrated in Fig. 4. In addition to the standard feature-based similarity used by AP, we consider also a topology-based property that reflects the knowledge encoded by the available network information for graph-structured data. To efficiently minimize the novel energy function, we perform a max-sum belief propagation over a factor graph where the data points are represented by variable nodes and the exemplars are represented by function nodes. Depending on the granularity of the desired clustering and the density of the network topology, the distribution of the identified exemplars over the network is expected to respect the constraints set for the optimization task but does not guarantee the presence of an exemplar in the topological neighborhood of every data point. As such, we propose a novel two-stage cluster assignment policy that, at a first stage, prioritizes the selection among neighbor exemplars and then relaxes the selection constraint to consider all the available exemplars if no exemplar exists within a specified distance from a given data point. This step is followed by a corrective label smoothing operation that uses graph coverings to generate more reliable cluster labels by looking at the direct neighborhood of every data point. Our extensive evaluation experiments on citation and social networks show the clear advantages of Geometric-AP over traditional approaches that lack the ability to leverage the network information independently from the feature-based knowledge.

The remainder of this article is organized as follows: In Section II, we briefly provide a general description of AP. In Section III, we introduce the new Geometric-AP algorithm. The new model is first described and its underlying rationale is discussed. Then the new message updates are derived using a max-sum belief propagation algorithm to optimize the redesigned net similarity. A comparative study between AP and Geometric-AP is conducted to show that the Geometric-AP algorithm can be viewed as a special case of AP that penalizes some clustering configurations under topological constraints. Sections IV and V report the experimental results on two citation networks and one social network, respectively. Section VI summarizes the insights gained from our analysis and provides some guidelines about the hyper-parameters tuning. The limitations of the proposed approach are also discussed with an emphasis on some potential future research directions. In Section VII we provide concluding remarks for this article.

II. Brief Review of Affinity Propagation

Affinity Propagation (AP) is a message passing algorithm that takes as input user-defined similarity measures for all data point pairs. Real-valued messages called *responsibility* and *availability* are iteratively exchanged between data points until a set of high-quality clusters gradually emerge around representative data points referred to as exemplars [15]. Based on the provided

input $s(i,j)$, AP subsequently exchanges the two types of messages between data points to decide which instance would serve as a good exemplar. The first message, called *responsibility* and denoted by $r(i,j)$, designates the message sent from point i to candidate exemplar point j. The second communicated message is called *availability* and is denoted by $a(i,j)$. The exchanged messages are defined and updated as follows:

$$r(i,j) \leftarrow s(i,j) - \max_{j's.t.j' \neq j} \{a(i,j') + s(i,j')\}. \tag{1}$$

$$a(i,j) \leftarrow \min \left\{ 0, r(j,j) + \sum_{i's.t.i' \notin \{i,j\}} \max\{0, r(i',j)\} \right\}. \tag{2}$$

Initially, all the availability messages are set to 0, except for the self availability, which is computed as follows:

$$a(j,j) \leftarrow \sum_{i's.t.i' \neq j} \max\{0, r(i',j)\}. \tag{3}$$

This ensures that the self availability of a given point is not inflated by higher responsibilities received from other points. In order to avoid numerical instabilities that may result from oscillating updates, an exchanged message m is damped as follows:

$$\mathrm{m}^{(t)} \leftarrow \lambda \mathrm{m}^{(t-1)} + (1-\lambda)\mathrm{m}^{(t)}, \tag{4}$$

where λ is the damping factor. Finally, at each iteration, we can determine the exemplar associated with each point by evaluating the following equation:

$$\texttt{exemplar}(i) = \arg\max_j \{a(i,j) + r(i,j)\}. \tag{5}$$

AP converges when the clustering configuration remains steady for a predefined number of iterations.

III. Geometric Affinity Propagation

Geometric Affinity Propagation (Geometric-AP) stems from the original formulation of AP where the clustering task has been viewed as a search over a wide set of valid configurations of the class labels $\mathbf{c} = (c_1, c_2, \ldots, c_N)$ for N data points [15]. Given a user-defined similarity matrix $[s_{ij}]_{N\times N}$, the search task has been defined in [15] as an optimization problem that aims at minimizing an energy function:

$$E(\mathbf{c}) = -\sum_{i=1}^{N} s(i, c_i), \tag{6}$$

where $s(i,c_i)$ is the similarity measure between data point i and its corresponding exemplar c_i. With $s(i,c_i)$ being a negative distance measure, minimizing $E(\mathbf{c})$ aims at assigning each data point to its nearest cluster exemplar.

Under valid configuration constraints, it has been shown in [15] that the optimization problem can be reformulated as the maximization of a net similarity $\mathcal{S}$, defined as:

$$\mathcal{S}(\mathbf{c}) = -E(\mathbf{c}) + \sum_{k=1}^{N} \delta_k(\mathbf{c})$$

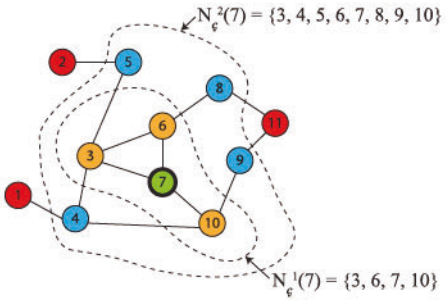


Fig. 1. Node neighborhood using `shortest_path` distance. Neighbors around node 7 are color-coded based on their corresponding `shortest_path` distance layer. Orange color represents directly adjacent nodes, blue color highlights nodes at 2 hops from node 7, and red color labels the remaining nodes in the graph.

$$= \sum_{i=1}^{N} s(i, c_i) + \sum_{k=1}^{N} \delta_k(\mathbf{c}), \tag{7}$$

where $\delta_k(\mathbf{c})$ is a penalty term expressed as:

$$\delta_k(\mathbf{c}) = \begin{cases} -\infty, & \text{if } c_k \neq k \text{ but } \exists\, i:\ c_i = k, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

In this formulation, a hard constraint has been set on top of configurations where a non-exemplar data point is being selected by another point as an exemplar. In other words, no data point is allowed to be elected by other data points as an exemplar, unless it has identified itself as an exemplar too. In Geometric-AP, for valid association between any pair of data points (i,k) that satisfies $c_i = k$, we further require that $k \in N_{\mathcal{G}}^{\tau}(i)$ where $N_{\mathcal{G}}^{\tau}(i)$ is the topological neighborhood of diameter τ with respect to graph $\mathcal{G}$ for the point i defined as:

$$N_{\mathcal{G}}^{\tau}(i) = \{x : \text{distance}_{\mathcal{G}}(x,i) \leq \tau\}, \tag{9}$$

where $\text{distance}_{\mathcal{G}}$ represents a topological distance with respect to graph $\mathcal{G}$. This means that no data point is allowed to elect an exemplar outside of its topological neighborhood that is specified by $N_{\mathcal{G}}^{\tau}$.

Fig. 1 illustrates the node neighborhood using a `shortest_path` distance.

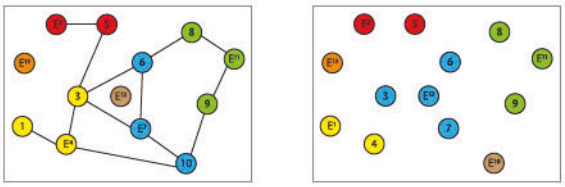
A. The Geometric Model

By implementing the network constraints, we aim at avoiding the configurations where a data point i chooses k as its exemplar (i.e., $c_i = k$) while $k \notin N_{\mathcal{G}}^{\tau}(i)$. Towards this end, we amend the penalty term $\delta_k(\mathbf{c})$ in (8) to a new penalty term $\gamma_k(\mathbf{c})$ that takes the form:

$$\gamma_k(\mathbf{c}) = \begin{cases} -\infty, & \text{if } c_k \neq k \text{ but } \exists\, i:\ c_i = k, \\ -\infty, & \text{if } \exists\, i:\ c_i = k \text{ but } k \notin N_{\mathcal{G}}^{\tau}(i), \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

The net similarity $\mathcal{S}$ in (7) becomes:

$$\mathcal{S}(\mathbf{c}) = \sum_{i=1}^{N} s(i, c_i) + \sum_{k=1}^{N} \gamma_k(\mathbf{c}). \tag{11}$$



(a) Geometric-AP (b) AP

Fig. 2. Geometric-AP versus AP. Identified clusters are color-coded. The shared preferences for AP and Geometric-AP are preselected to get the same number of clusters. The node features are the plane coordinates. Geometric-AP has been launched with the `shortest_path` distance and a neighborhood threshold of 2 ($\tau = 2$). Cluster exemplars are labeled by E^{ID}.

In addition to maximizing the within-cluster feature-based similarity, this new formulation intuitively maximizes the within-cluster topological similarity. As a result, the optimization task jointly searches for valid configurations that account for both feature-based and network-based similarities. From this perspective, Geometric-AP can be viewed as a more constrained special case of AP. We straiten the search space of proximal exemplars for a given point i to the local neighborhood $N_{\mathcal{G}}^{\tau}(i)$. When no exemplar exists in $N_{\mathcal{G}}^{\tau}(i)$, the point i is allowed to select among the full list of emerged exemplars. Clearly, this assignment policy may raise some misassignments when the exemplar selection occurs outside of the local neighborhood. To remedy this deficiency, we smooth the labels throughout the network using adjacency majority voting. Indeed, the network adjacency of any given point i with respect to a graph $\mathcal{G}$, denoted by $\mathcal{A}_{\mathcal{G}}(i)$, can be viewed as an α-cover of the reduced graph formed by i and $\mathcal{A}_{\mathcal{G}}(i)$. For instance, in [32] the authors provided a label selection strategy using graph coverings and have derived an upper-bound expression for the error committed by majority voting in binary labeled graphs.

Fig. 2 illustrates the clustering properties of Geometric-AP as compared to AP when applied to one synthetic dataset. Obviously, Geometric-AP is more robust against outliers and generates better connected modules in the network. In contrast, AP is hypersensitive to an unconnected vertex (node 12) as it only relies on feature similarities. Additionally, the exemplars identified by Geometric-AP occupy more centric locations in the network when compared to the ones selected by AP.

B. Topological Neighborhood

Geometric-AP greatly depends on the neighborhood function $N_{\mathcal{G}}^{\tau}$ defined in (9). In order to probe the effect of the topological distance "distance$_{\mathcal{G}}$" on the performance of Geometric-AP, we consider throughout this article three widely used topological distance metrics. We evaluate the performance of our geometric model using the Jaccard, cosine, and shortest path distances. Unless otherwise stated, we consider in this study a network information in the form of an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of vertices of size N mapped to the observed data points in the feature space (i.e., every data point is mapped to a unique vertex in $\mathcal{G}$). $\mathcal{E}$ represents the set of undirected edges in the graph $\mathcal{G}$.

1) Jaccard Distance: The Jaccard distance is derived from the Jaccard index defined for two sets A and B as:

$$\rho(A, B) = \frac{|A| \cap |B|}{|A| \cup |B|}. \tag{12}$$

From a topological viewpoint, we characterize every vertex $V \in \mathcal{V}$ by an M-dimensional binary vector $V = (v_1, v_2, \ldots, v_M)$ such that $v_{i_{(i=1..M)}} = 1$ if vertex V and V_i are connected in $\mathcal{G}$ and 0 otherwise. The Jaccard distance between two vertices $V_1 = (v_{11}, v_{12}, \ldots, v_{1M})$ and $V_2 = (v_{21}, v_{22}, \ldots, v_{2M}) \in \mathcal{V}$ is then defined as:

$$\begin{aligned} \text{distance}_{\mathcal{G}}^{Jaccard}(V_1, V_2) &= 1 - \rho(V_1, V_2) \\ &= \frac{C_{1,0} + C_{0,1}}{C_{1,0} + C_{0,1} + C_{1,1}}, \end{aligned} \tag{13}$$

where $C_{i,j}$ is the number of positions $k \in [1..M]$ in which $v_{1k} = i$ and $v_{2k} = j$.

2) Cosine Distance: The cosine distance between two vertices $V_1 = (v_{11}, v_{12}, \ldots, v_{1M})$ and $V_2 = (v_{21}, v_{22}, \ldots, v_{2M}) \in \mathcal{V}$ is given by:

$$\text{distance}_{\mathcal{G}}^{cosine}(V_1, V_2) = 1 - \frac{V_1.V_2}{\sqrt{\sum_{k=1}^{M} v_{1k}^2} \cdot \sqrt{\sum_{k=1}^{M} v_{2k}^2}}. \tag{14}$$

3) Shortest Path Distance: The shortest path distance between two vertices V_1 and $V_2 \in \mathcal{G}$ is universally defined as the shortest sequence of edges in $\mathcal{E}$ starting at vertex V_1 and ending at vertex V_2. To fully determine the neighborhood function $N_{\mathcal{G}}^{\tau}$ we need only to know about the existence of shortest paths with specified lengths but not the full sequence of edges. This observation leads to an efficient implementation of the neighborhood function $N_{\mathcal{G}}^{\tau}$ based on the shortest path distance. Indeed, for any vertex i, the determination of $N_{\mathcal{G}}^{\tau}(i)$ is straightforward if we observe that the νth power of a graph $\mathcal{G}$ is also a graph with the same set of vertices as $\mathcal{G}$ and an edge between two vertices if and only if there is a path of length at most ν between them.If we denote by $\mathcal{A}(\mathcal{G})$ the adjacency matrix of graph $\mathcal{G}$ and by $\mathcal{A}(\mathcal{G}^{\nu})$ the adjacency matrix of graph $\mathcal{G}^{\nu}$, the entries of $\mathcal{A}(\mathcal{G}^{\nu})$ are derived using an indicator function as follows:

$$\mathcal{A}(\mathcal{G}^{\nu}) = \mathbb{1}_{\left[\sum_{i=1}^{\nu} \mathcal{A}(\mathcal{G})^i\right]}, \tag{15}$$

where the indicator function $\mathbb{1}$ of matrix X with entries x_{ij}, is a matrix of the same dimension as X and entries defined by: $\mathbb{1}_X[i, j] = 1$ if $x_{ij} \neq 0$ and $\mathbb{1}_X[i, j] = 0$ otherwise.

C. Optimization

Similar to the original AP algorithm, the optimization of the objective function introduced in Geometric-AP is NP-hard. In order to estimate the optimal label configuration we follow similar derivation as the one introduced in [15]. We solve this optimization problem using max-sum belief propagation over the factor graph depicted in Fig. 3. We note that the function node in Fig. 3 is different from the one presented in [15] as it leverages the network information to account for the topological similarity.

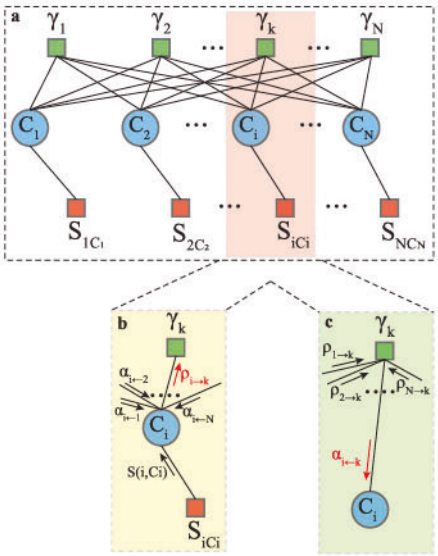


Fig. 3. Factor graph for Geometric-AP.

In the max-sum belief propagation algorithm, a bipartite message communication between two types of nodes is conducted with an alternation between summation and maximization steps as illustrated in Fig. 3(a). The first type of nodes is called variable node and it sums up the received messages from all second type nodes, called function nodes, other than the one receiving the message (Fig. 3(b)). Likewise, every function node maximizes its value over all the variables except the variable the message is being sent to (Fig. 3(c)). We next provide the set of derived message updates that govern Geometric-AP.

1) Message Updates: With analogy to AP, the message sent from variable node c_i to function node γ_k sums together all the messages received from the remaining function nodes. As shown in Fig. 3(b), this message is denoted by $\rho_{i\to k}$ and takes the form:

$$\rho_{i\to k}(c_i) = s(i,c_i) + \sum_{k':k'\neq k} \alpha_{i\leftarrow k'}(c_i). \tag{16}$$

Similarly, the message sent from function node γ_k to variable node c_i computes the maximum over all variable nodes except c_i (Fig. 3(c)) and can be given by:

$$\alpha_{i\leftarrow k}(c_i) = \max_{(c_1,c_2,\ldots,c_{i-1},c_{i+1},\ldots,c_N)} \left[\gamma_k(c_1,c_2,\ldots,c_{i-1},\mathbf{c_i},c_{i+1},\ldots,c_N) + \sum_{i':i'\neq i} \rho_{i'\to k}(c_{i'})\right]. \tag{17}$$

Using a set of mathematical simplifications utilized in [15] we derive two message updates that we also call *responsibility* and *availability*. Here, the responsibility message, denoted by $r(i,k)$, replaces the message $\rho_{i\to k}$ and the availability message designated by $a(i,k)$ substitutes the message $\alpha_{i\leftarrow k}$ as shown in Fig. 3. Ultimately, the simplified messages are given by (18) and (19) shown at the bottom of this page.

$$r(i,k) = s(i,k) - \max_{j:j\neq k}[s(i,j) + a(i,j)]. \tag{18}$$

The detailed derivation of message updates for Geometric-AP is provided in Section I of the supplemental material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TKDE.2023.3237630. Obviously, the responsibility message remained unchanged as compared to AP. However, the availability message has accommodated the network constraints and contained an additional update that takes into account the network neighborhood between data points and potential exemplars. This is reflected in the message update in (19) since the availability message remains unchanged when the candidate exemplar falls within the topological neighborhood of the communicating data point but takes a new expression whenever the potential exemplar is located outside the network proximity of the data point of interest. In the next section we discuss an updated assignment policy that consolidates the underlying concepts of the derived messages.

2) Assignment of Clusters: At any given iteration of AP, the value of a variable node c_i can be estimated by summing together all messages that c_i receives. Subsequently, the argument that maximizes these incoming messages, denoted by $\hat{c}_i$, will be a good estimate for c_i [15].

$$\hat{c}_i = \arg\max_j [a(i,j) + s(i,j)]. \tag{20}$$

In Geometric-AP, this rule is amended to be consistent with the joint similarity criteria respected during the derivation of the message updates. Indeed, Geometric-AP prioritizes the assignment of data points to the closest exemplar that lies within the local neighborhood of each data point. As the number of emerging exemplars is automatically determined by the algorithm, some proximal exemplars may breach the topological constraint. To remedy this possible deficiency, we prioritize at a first stage the selection among close exemplars that fall within the topological sphere determined by $N_{\mathcal{G}}^{\tau}$. As Geometric-AP does not guarantee that at least one exemplar emerges in the network neighborhood of every data point, we allow at a second stage a more lenient selection among all available exemplars. The new updated rule takes then the form: (21) shown at the bottom of the next page.

$$a(i,k) = \tilde{\alpha}_{i\to k}(c_i = k) = \begin{cases} \sum_{i':i'\neq k} \max(0, r(i,'k)), & \text{if } k = i, \\ \min\left(0, r(k,k) + \sum_{i':i'\notin\{i,k\}} \max(0, r(i,'k))\right), & \text{if } k \neq i \;\&\; k \in N_{\mathcal{G}}^{\tau}(i), \\ -\max\left(0, r(k,k) + \sum_{i':i'\notin\{i,k\}} \max(0, r(i,'k))\right), & \text{if } k \neq i \;\&\; k \notin N_{\mathcal{G}}^{\tau}(i). \end{cases} \tag{19}$$

3) Label Smoothing: The misassignments that may happen in Geometric-AP, particularly when the exemplar selection takes place outside of the local neighborhood, reduces the performance of the carried out clustering. To mitigate this issue we adopt a label smoothing strategy that has been widely used in label selection on graphs. To this end, we consider graph coverings that use α-cover sets to label the remaining nodes in the graph by majority vote [32]. By definition, we say that a set S α-covers a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ if $\forall i \in \mathcal{V}$ either $i \in S$ or $\sum_{j \in S} W_{ij} > \alpha$ where W_{ij} denotes the weight on the edge between vertex i and j. For unweighted graphs W_{ij} takes a binary value. Realistically, a vertex V in $\mathcal{G}$ can be labeled efficiently by majority vote if some voting nodes are adjacent to V w.r.t $\mathcal{G}$. From this perspective, we target all adjacent voters and we consider the network adjacency of any given point i w.r.t $\mathcal{G}$, denoted by $\mathcal{A}_{\mathcal{G}}(i)$, to form a reduced graph formed by i and $\mathcal{A}_{\mathcal{G}}(i)$. In this reduced graph, for $\alpha = 1$, $\mathcal{A}_{\mathcal{G}}(i)$ can be viewed as an α-cover for $\mathcal{A}_{\mathcal{G}}(i) \cup i$. In our work, we perform an inclusive label smoothing throughout the full network in a way to prevent Geometric-AP from being overconfident. We propose the following label smoothing policy:

$$\hat{c}_i = \arg\max_k \left[\sum_{j \in \mathcal{A}_{\mathcal{G}}(i)} \delta\left(c_j = c_k\right) \right], \tag{22}$$

where $\delta(.)$ is the Dirac delta function, such that the sum in (22) counts the number of vertices in $\mathcal{A}_{\mathcal{G}}(i)$ that have class c_k. In order to comprehensively handle all vertices in $\mathcal{G}$, including disconnected nodes, Geometric-AP counts the vote of the vertex i as well.

D. Comparison to Affinity Propagation

Geometric-AP leverages the available network information by implementing a more constrained optimization problem as compared to AP. This implementation assumes that significant clusters are jointly compact in two different domains that are the node-feature domain and the network topological domain. Under this assumption, Geometric-AP is expected to boost the clustering performance when the information carried by the node features and the network topology about the ground-truth clusters is consistent. From this standpoint, the network information can be viewed as a chaperone for the clustering task to achieve a more significant partitioning of the data by avoiding some local optimum traps. To further elucidate this notion, we rewrite the message updates in (19) using the identity:

$$x - \max(0, x) = \min(0, x), \tag{23}$$

which leads to the expression given in (24) shown at the bottom of this page. This new result outlines the difference between Geometric-AP and AP as a penalty term deducted from the availability message sent from the potential exemplar k to the data point i when $k \notin N_{\mathcal{G}}^{\tau}(i)$. The expression of the penalty term is given by:

$$r(k,k) + \sum_{i':i' \notin \{i,k\}} \max\left(0, r\left(i,' k\right)\right). \tag{25}$$

Except the time required to compute the neighborhood function $N_{\mathcal{G}}^{\tau}$, the expression provided in (24) sets the computational complexity of Geometric-AP to $\mathcal{O}(N^2)$ since the expression of the penalty term is already computed and is reusable without any additional computational cost. All these advantages make the implementation of the proposed Geometric-AP as efficient as that of AP while keeping the benefits carried by the network information.

IV. Unsupervised Document Clustering

We thoroughly study, in this section, the improvement of Geometric-AP with reference to AP in unsupervised document classification on two benchmark citation networks, that are the *cora* dataset [33] and the *citeseer* dataset [34]. Additionally, we select and perform a variety of clustering methods that span many state-of-the-art clustering approaches for comparison purposes. Our findings show that Geometric-AP consistently outperforms AP on the studied datasets and exhibits high competitiveness with other long-standing and popular methods.

$$\hat{c}_i = \begin{cases} j: \begin{bmatrix} j \in N_{\mathcal{G}}^{\tau}(i) \\ \& \\ c_j = j \end{bmatrix}, & \text{if} \begin{bmatrix} \exists k \in N_{\mathcal{G}}^{\tau}(i) \\ : c_k = k \end{bmatrix}, \\ j, & \text{otherwise.} \end{cases} \underset{j}{\operatorname{argmax}} \left[a\left(i,j\right) + s\left(i,j\right)\right]. \tag{21}$$

$$a(i,k) = \tilde{\alpha}_{i \to k}\left(c_i = k\right) = \begin{cases} \sum_{i':i' \neq k} \max\left(0, r\left(i,' k\right)\right), & \text{if } k = i, \\ \min\left(0, r(k,k) + \sum_{i':i' \notin \{i,k\}} \max\left(0, r\left(i,' k\right)\right)\right), & \text{if } k \neq i \;\&\; k \in N_{\mathcal{G}}^{\tau}(i), \\ \min\left(0, r(k,k) + \sum_{i':i' \notin \{i,k\}} \max\left(0, r\left(i,' k\right)\right)\right) - \left[r(k,k) + \sum_{i':i' \notin \{i,k\}} \max\left(0, r\left(i,' k\right)\right)\right], & \text{if } k \neq i \;\&\; k \notin N_{\mathcal{G}}^{\tau}(i). \end{cases} \tag{24}$$

A. Methods and Settings

The list of clustering methods selected to benchmark Geometric-AP along with their tuned hyper-parameters are detailed as follows:

1) Exemplar-based clustering methods. The selected competing methods are kmedoids [4] and AP [15]. We denote the convergence parameters of AP and Geometric-AP by $\max_{\text{iter}}$, $\text{conv}_{\text{iter}}$, and λ to designate the maximum number of iterations, the number of iterations for convergence, and the message damping factor, respectively. We choose values reported to guarantee high convergence rates [35]. $\max_{\text{iter}} = 1000$, $\text{conv}_{\text{iter}} = 100$, and $\lambda = 0.9$. kmedoids relates to the kmeans [3] algorithm but it identifies the medoid of each cluster by minimizing the sum of distances between the medoid and data points instead of sum-of-squares. Unlike centroids, medoids are selected from the existent data points.
2) Centroid-based clustering. The most popular method, that is kmeans [3], is performed. kmeans is run 1000 times with random centroid seeds and the best performance is reported.
3) Structural clustering. The clustering method *spectral-g* [36], which takes the network adjacency matrix as the similarity matrix, is selected and compared. In *spectral-g*, the eigenvectors of the graph Laplacian are computed and the kmeans algorithm is used to determine the clusters. The assignment process is repeated 1000 times with random initialization and the best result is recorded.
4) Hierarchical clustering. To mimic the operating mode of AP where initially all data points can be exemplars, we select a bottom-up hierarchical clustering method that is the hierarchical agglomerative clustering (HAC) [37]. To prioritize compact clusters with small diameters we further consider the complete linkage criterion for merging similar clusters.
5) Model-based clustering. We also test a Gaussian mixture model (GMM) [38] method that utilizes the Expectation-Maximization (EM) algorithm to fit a multi-variate Gaussian distribution per cluster. Initially, the probability distributions are centered using kmeans and then EM is used to find local optimal model parameters using full covariances. The mixture model is employed afterwards to assign data points to the classes to maximize the posterior density. 100 random restarts are performed and the best performance is reported.
6) Variational inference clustering. We perform a Bayesian variational inference clustering by fitting a Gaussian mixture model with an additional regularization from a prior Dirichlet process distribution (DPGMM) [39]. Similar to GMM, 100 random restarts with full covariances are performed and the best result is reported.
7) Density-based clustering. We test a spatial density-based clustering algorithm (DBSCAN) that identifies core data points of high density and expands from them to obtain the target clustering. To attain the desired number of clusters, we leniently reduce the required size of core points' neighborhood and we adjust the distance threshold within each cluster accordingly.
8) Graph-based clustering. We select a minimum spanning tree (MST) approach that identifies the desired clusters as sub-graphs that connect node members in a way to minimize the sum of the graph edges. The number of final clusters is controlled through a cutoff threshold.

Geometric-AP is implemented in 64-bit python 3.6.8 on a workstation (Windows 64 b, 2.8 GHz Intel Core i7-7700HQ CPU, 16 GB of RAM).

B. Similarity Metrics

Many methods have been devised in the past few decades to provide vector representations for textual data [31], [40]. Most popular representations that have been extensively used in the literature include the binary word vector and the term-frequency inverse-document-frequency (tf-idf) [40] representations. Distance measures that have been reported as congruent with these representations include the Euclidean, Manhattan, and cosine distances as reviewed in [31]. For unbiased comparison, we independently run the kmedoids algorithm using the aforementioned distances on the *cora* and *citeseer* datasets to predict the ground-truth class labels and we retain the distance measure that gives the best clustering result on each dataset. In an M-dimensional space, the considered distances between two data points $p = (p_1, p_2, \ldots, p_M)$ and $q = (q_1, q_2, \ldots, q_M)$ are defined as follows:

- Euclidean Distance: $dist_{Euc}(p,q) = \sum_{i=1}^{M} \sqrt{(p_i - q_i)^2}$.
- Manhattan Distance: $dist_{Man}(p,q) = \sum_{i=1}^{M} |p_i - q_i|$.
- Cosine Distance: $dist_{Cos}(p,q) = \frac{p.q}{\sqrt{\sum_{i=1}^{M}(p_i)^2}\sqrt{\sum_{i=1}^{M}(q_i)^2}}$.

Consequently, the off-diagonal elements of the similarity matrix $[s_{ij}]_{N\times N}$ between N data points are defined as:

$$s_{ij} = -dist_{metric}(i,j), \tag{27}$$

where $dist_{metric}$ refers to one of the previously discussed distance measures. The preference values s_{ii} are controlled over a range of values to generate different number of clusters.

C. Clustering Evaluations

In the absence of a unified criterion universally accepted for assessing clustering performance, many evaluation metrics have been proposed. The list of popular metrics include, but not limited to, average purity, entropy, and mutual information [41]. More recently, mutual information measures become accepted with appreciation by the research community as they provide a plausible evaluation of the information shared between the compared clusterings. We adopt three widely used performance measures as discussed in [42] that are: normalized mutual information (NMI), classification rate (CR), and macro F1-score (F1).

D. Performance Assessment Results

1) Cora Dataset: The *cora* dataset [33] contains 2708 machine learning papers from seven classes and 5429 links between

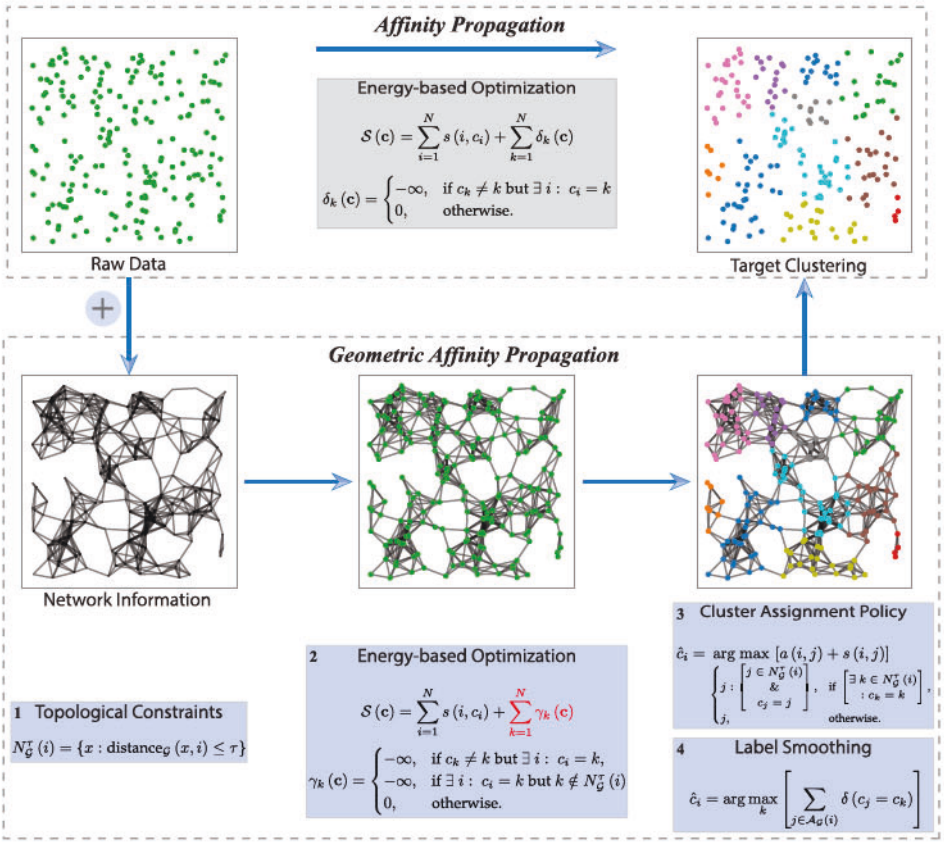


Fig. 4. Geometric-AP leverages the network information to better unveil the hidden data structures. The clustering task is reformulated as an energy-based optimization problem under additional topological constraints imposed by the available network information.

them. The links indicate a citation relationship between the papers. Each document is represented by a binary vector of 1433 dimensions marking the presence of the corresponding word. The documents in *cora* are short texts extracted from titles and abstracts where the stop words and all words with document frequency less than 10 are removed [43]. Stop words include non-informative words like articles and prepositions and are filtered out to avoid inflating the dimensions. Each document in *cora* has on average 18 words and the network is regarded, in the context of this work, as an undirected graph. After applying the selection procedure described in Section IV-B, the most appropriate similarity metric on the *cora* dataset has been identified as the negative Euclidean distance.

Next, we plot in Fig. 5(a), (b), and (c) the clustering results of Geometric-AP with different neighborhood functions $N_{\mathcal{G}}^{\tau}$ as a function of τ when the ground-truth classes are being considered. In this experiment we aim at tuning $N_{\mathcal{G}}^{\tau}$ defined in (9) by identifying the best topological distance "distance$_{\mathcal{G}}$" and the optimal threshold value τ that lead to the top clustering result w.r.t NMI, CR, and F1. In case of conflicts or ties, the reference metric for identifying the optimal threshold is always the NMI and the smallest optimal threshold is retained. For each topological distance, we run the algorithm with τ ranging from 1 to 5 for the shortest path distance and from 0.5 to 0.9 for the Jaccard and cosine metrics when the actual classes are being used, i.e., 7 in the case of *cora*. As illustrated in Fig. 5(a), the optimal clustering results are obtained using the `shortest_path` distance with a neighborhood threshold $\tau = 3$. In the remainder of this discussion about the *cora* dataset the neighborhood function $N_{\mathcal{G}}^{\tau}$ will be defined as follows:

$$N_{\mathcal{G}}^{\tau}(i) = \{x : \texttt{shortest\_path}(x, i) \leq 3\}. \tag{28}$$

We show in Fig. 5(d), (e), and (f), respectively, the NMI, CR, and F1 histograms for the different algorithms. Clearly, Geometric-AP significantly outperforms all other algorithms including non-exemplar-based methods on the three evaluation metrics. This result suggests that the network information comprises a substantial knowledge about the structure of the ground-truth categories. Additionally, the edge distribution captured by the neighborhood function $N_{\mathcal{G}}^{\tau}$ in (28) seems to be highly compatible with the identified exemplars as it increases the affinity of cluster members to the most appropriate representatives.

In Fig. 5(g), (h), and (i) we plot NMI, CR, and F1 as functions of K, the number of identified clusters. Detecting variable K is possible by calibrating the self-preferences $[s_{ii}]_{N \times N}$ over a range of values in AP and Geometric-AP. Remaining clustering methods generate the desired number of clusters by specifying the preferred number of cluster components in initialization. Whenever kmeans is used, the simulations are repeated 1000 times with random restarts and the best performance is plotted. Henceforward, the aforementioned setup is used when reporting the NMI, CR, and F1 values as functions of number of clusters

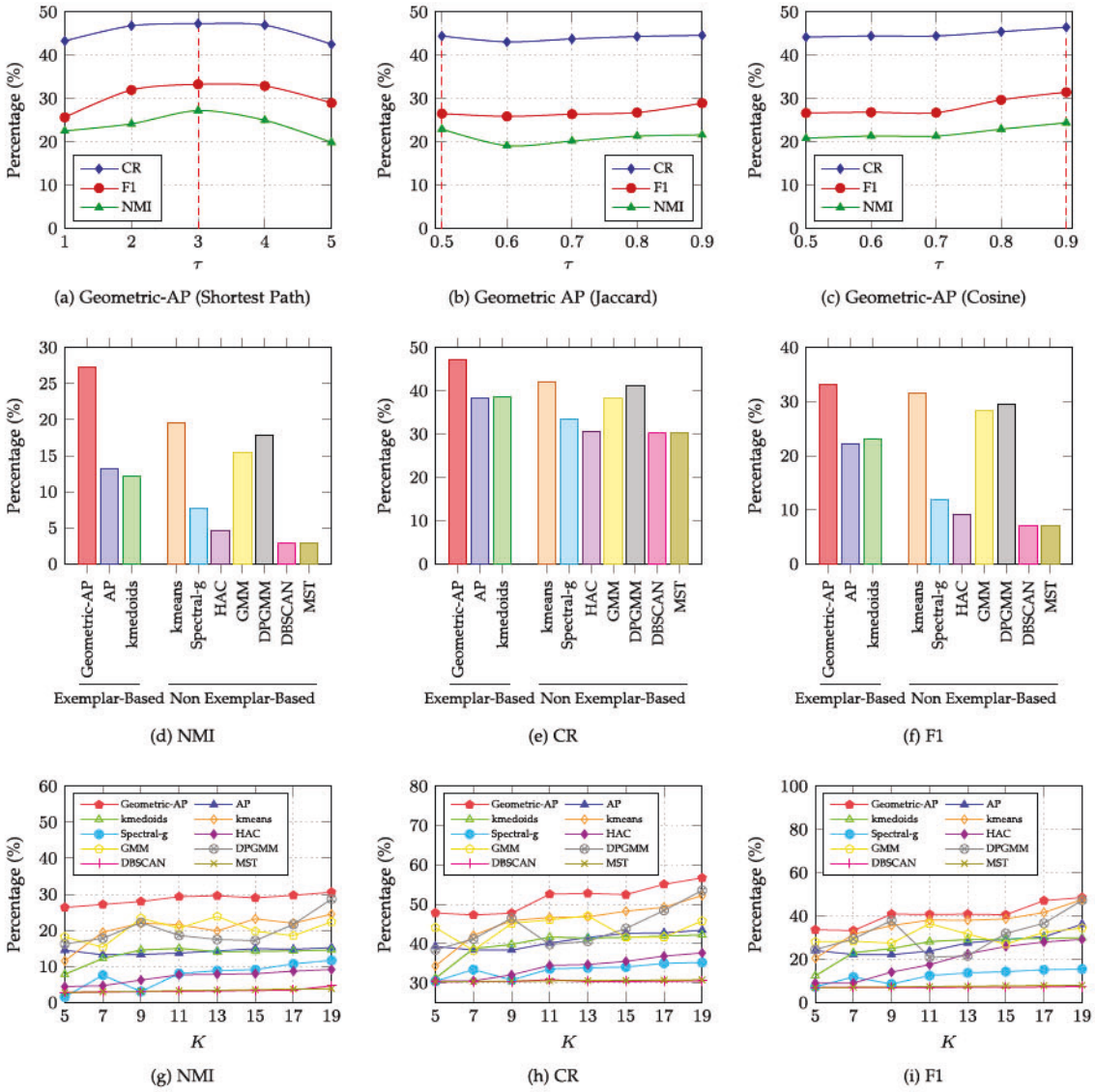


Fig. 5. Hyperparameter tuning and evaluation results on the *cora* dataset. (a-c) The three distance metrics with variable threshold values are tested to predict the ground-truth categories, i.e., 7. By referring to the NMI metric, the optimal clustering results are obtained using the `shortest_path` distance and $\tau = 3$. Plot (d-f) NMI, CR, and F1 evaluation metrics are reported for Geometric-AP and the rest of benchmark algorithms. (g-i) Plots of evaluation metrics as function of K (number of identified clusters).

K. The figures show that Geometric-AP consistently outperforms other methods by a significant amount w.r.t NMI, CR, and F1 when the number of identified clusters spans the number of ground-truth categories.

2) Citeseer Dataset: The *citeseer* dataset [34] contains 3312 labeled publications spread over six classes with 4732 links between them. The links between documents indicate a citation relationship and each paper is represented by a binary word vector of dimension 3703 after stemming and removing stop words. Similar to *cora*, words with document frequency less than 10 are removed. On average, each document in *citeseer* has 32 words [43] and we harness the provided network information as an undirected graph. The metric selection procedure discussed in Section IV-B yields to the negative cosine distance as the best similarity measure on *citeseer* w.r.t our setup. Plots in Fig. 6(a), (b), and (c) illustrate the clustering results of Geometric-AP with different neighborhood functions $N_{\mathcal{G}}^{\tau}$ as a function of τ when the ground-truth classes are being predicted. This hyper-parameter tuning step is crucial and should be performed for any studied dataset. In *citeseer*, parameter tuning serves to identify the best neighborhood function $N_{\mathcal{G}}^{\tau}$ defined in (9) by searching for the finest topological distance "distance$_{\mathcal{G}}$" and the most adequate threshold value τ that engender the highest clustering result w.r.t NMI, CR, and F1. Deterministically, the optimal threshold value corresponds to the one that gives the largest NMI. For each topological distance, Geometric-AP is run with τ ranging from 1 to 7 for the shortest path distance and from 0.5 to 0.9 for the Jaccard and cosine metrics. The reference task for tuning the model parameters is the prediction of the true class labels, i.e., 6 in the case of *citeseer*. Fig. 6(a) shows that the `shortest_path` distance should be considered with a neighborhood threshold $\tau = 5$. Thereafter, the neighborhood

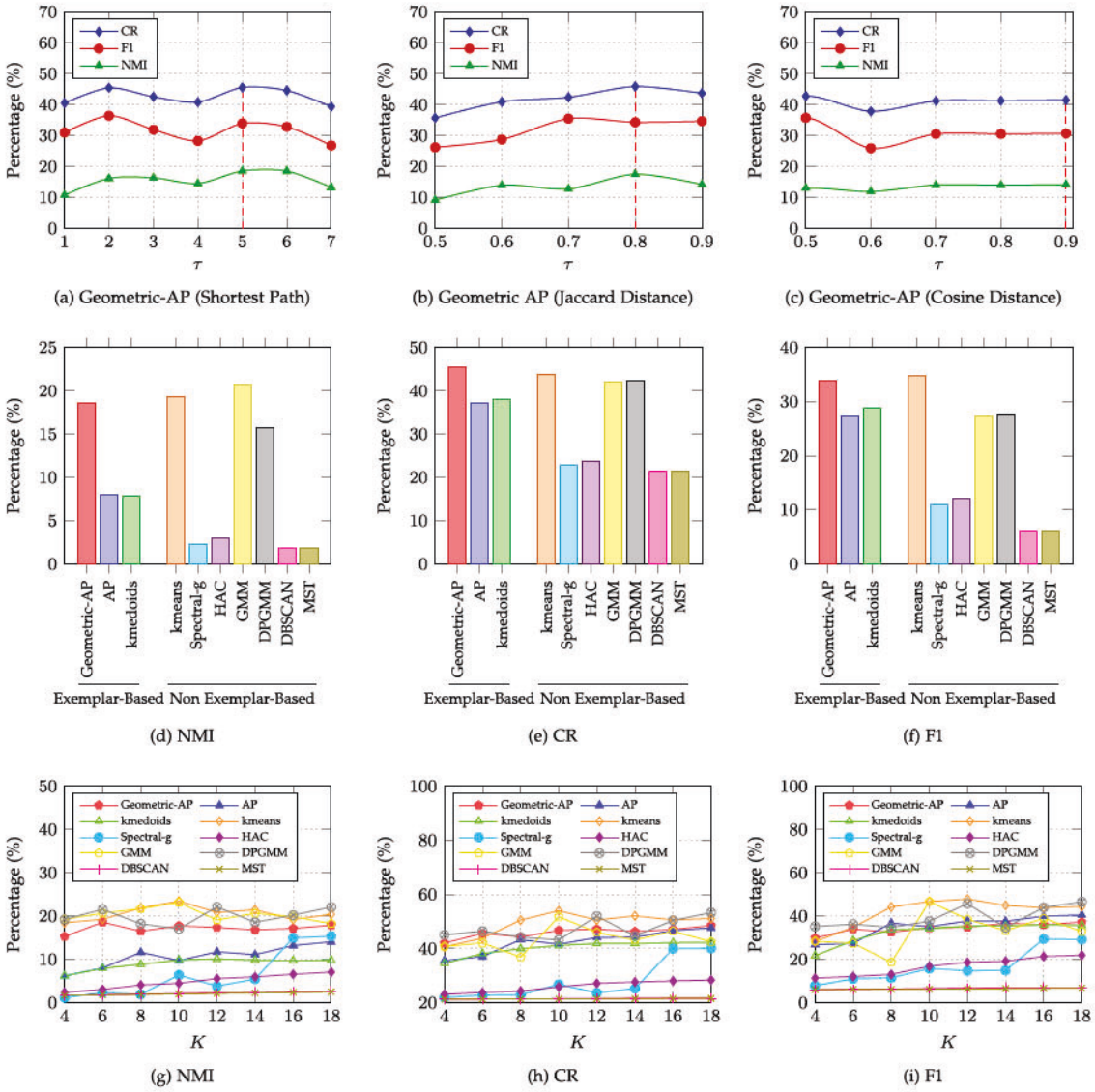


Fig. 6. Hyperparameter optimization and evaluation results on the *citeseer* dataset. (a-c) Three topological distances are tested over a range of threshold values to predict the ground-truth categories, i.e., 6. By consulting the NMI measure, the optimal clustering results are obtained using `shortest_path` distance and $\tau = 5$. (d-f) NMI, CR, and F1 evaluation metrics are reported for the set of tested algorithms. (g-i) Plots of evaluation metrics as functions of K(number of identified clusters).

function $N_{\mathcal{G}}^{\tau}$ is set as follows:

$$N_{\mathcal{G}}^{\tau}(i) = \{x : \texttt{shortest\_path}(x, i) \leq 5\}. \quad (29)$$

It should be noticed that, on the *citeseer* dataset the optimal neighborhood function encompasses larger diameter ($\tau = 5$) as compared to *cora* dataset where the best τ found to be 3. The main reason is that the network in *cora* dataset is more dense (network density of $14.812{*}10^{-4}$) and has higher average node degree (4) while the *citeseer* dataset has a network density of $85.179\ {*}10^{-5}$ and an average node degree equal to 2 [44]. As expected, increasing sparsity in the network usually compels Geometric-AP to delve deeper into the network to locate the best clustering structure.

We plot in Fig. 6(d), (e), and (f), respectively, the NMI, CR, and F1 histograms for the different algorithms. Like on *cora*, Geometric-AP significantly outperforms its counterparts AP and kmedoids and still behaves comparably to other top performing methods.

In Fig. 6(g), (h), and (i) we plot NMI, CR, and F1 as functions of K. The simulations confirm that Geometric-AP consistently outperforms exemplar-based methods and generates comparable results when confronted with other methods. A comparative analysis of the results obtained on *cora* and *citeseer* stipulates that the increase in clustering performance harvested from the network information diminishes as the true categories become more spread over the network. This observation is consistent with findings previously summarized in [31]. Additionally, the `shortest_path` distance has been identified on both datasets as the best distance measure. This is explained by the high flexibility offered by this metric to the potential exemplars as they are allowed to declare availability to non-neighbor nodes and data points at more than 2 hops in the network. In contrast, Jaccard

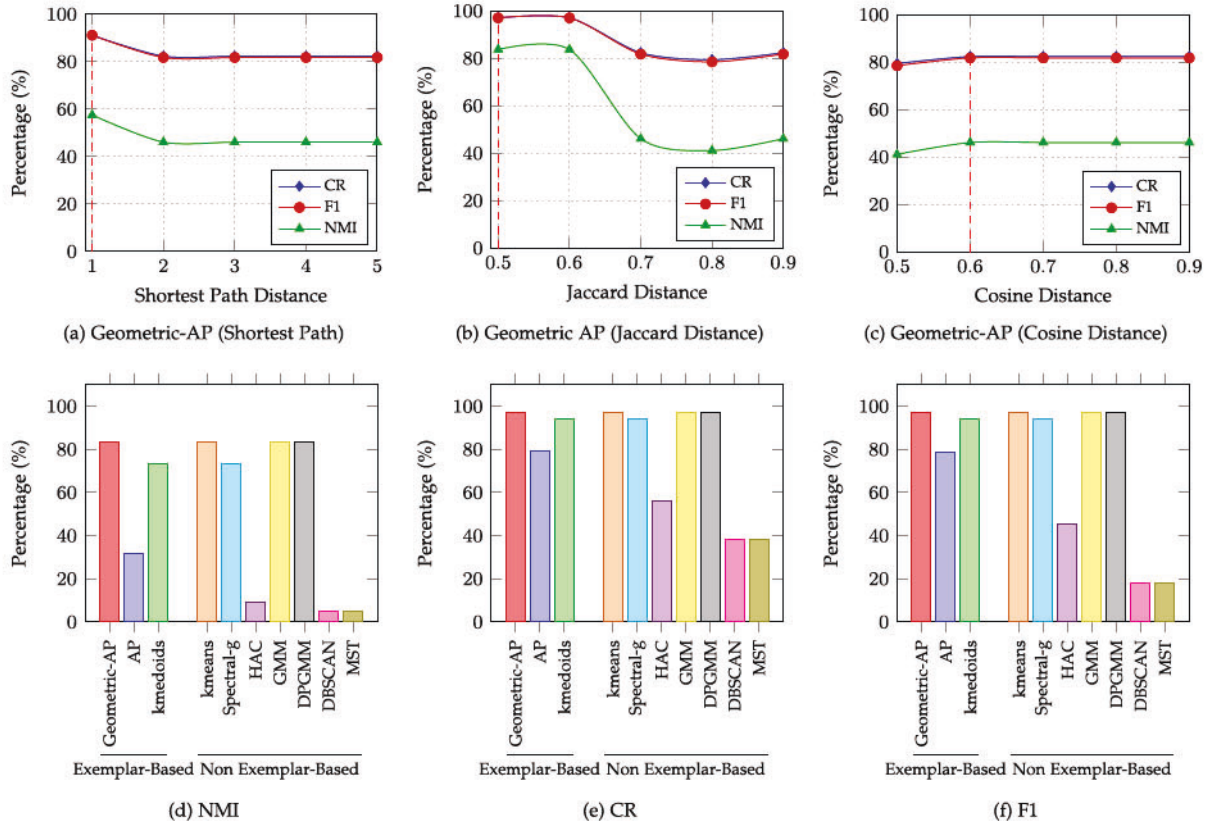


Fig. 7. Hyperparameter tuning and evaluation results on the *Zachary's Karate Club* network using the club split as ground-truth classes. (a-c) Selected topological distances are optimized over a range of threshold values to predict the ground-truth categories, i.e., 2. By referring to the NMI measure, the optimal clustering results are produced via `Jaccard` distance and $\tau = 0.5$. Plot (d-f) NMI, CR, and F1 evaluation metrics are reported for the different tested algorithms.

and cosine distances quantify only the shared neighborhood between two nodes in the network and as such the exemplars are unable to communicate their availability to nodes far more than 2 hops in the graph. Figs. 5(a) and 6(a) demonstrate that declaring availability to fewer data points than the optimal number engenders scarcity in the communication required to identify good clusters. Also, advertising the availability of exemplars in a broad manner introduces an additional noise that deteriorates the identification of good exemplars and misleads the search for valid label configurations.

V. Clustering of Social Networks

A. Zachary's Karate Club

The Zachary's karate club [45] is a social network of a university karate club that was monitored for two years. The network contains 34 members and 78 links between them. The links document the interaction between members outside the club. During the study, a conflict between the instructor and the administrator arose, which resulted in the split of the club into two sets. Each club member is represented by a binary vector of 34 dimensions indicating the interaction outside the club with other members. In the remaining of this section we evaluate the ability of Geometric-AP and other algorithms in retrieving the correct split of the club members. Also, we consider an additional 4-class partition obtained by modularity-based clustering [46] and we assess the clustering performance w.r.t these pseudo ground-truth labels.

Indeed, modularity has been first introduced in [47] as a quality measure for graph clustering. Thenceforth, it has attracted considerable research attention and becomes widely accepted as a quality index for graph clustering. By considering the additional modularity-based classes we aim at analyzing the power of Geometric-AP in sensing the modularity within graph-structured datasets. We note that the metric selection procedure presented in Section IV-B for both ground-truth class labels (club split and modularity-based classes) identifies the optimal similarity measure as the negative cosine distance.

1) Club Split Clusters: Similar to *cora* and *citeseer* datasets, we initially start by searching for the optimal topological distance and its corresponding threshold. Fig. 7(a), (b), and (c) shows that the `Jaccard` distance with a threshold value of 0.5 performs the best in retrieving the club split. The neighborhood function is then given by:

$$N_{\mathcal{G}}^{\tau}(i) = \{x : \texttt{Jaccard}(x, i) \leq 0.5\}. \tag{30}$$

As illustrated in Fig. 7(d), (e), and (f), Geometric-AP consistently outperforms its counterparts of exemplar-based methods and performs comparably to other top performing algorithms such as kmeans and Gaussian mixture models. While the standard AP failed to compete in recovering the actual split and

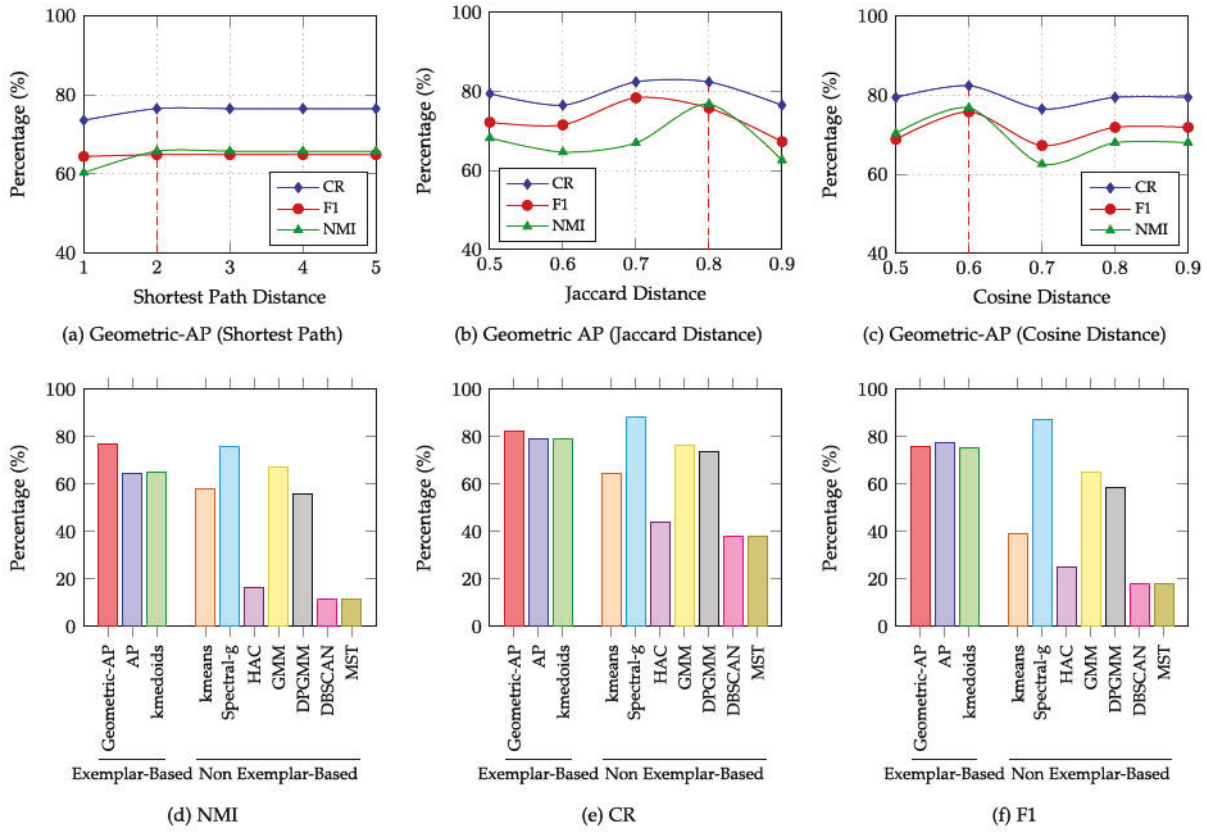


Fig. 8. Hyperparameter optimization and evaluation results on the *Zachary's Karate Club* dataset using the modularity-based classes as ground-truth labels. (a-c) Distance metrics are evaluated with variable threshold values to predict the ground-truth categories, i.e., 4. By taking the NMI measure as reference, the optimal clustering results are obtained with the `Jaccard` distance and $\tau = 0.8$. (d-f) NMI, CR, and F1 evaluation metrics are reported for the different tested algorithms.

misassigned 7 members, Geometric-AP has successfully identified the two sets with only one misclassified member. Obviously, our geometric model boosts the clustering accuracy with respect to all evaluation metrics with the largest enhancement being recorded for NMI, which has increased by 52% as compared to the standard AP.

2) Modularity-Based Clustering: Likewise, for the three topological distances being tested, the NMI, CR, and F1 scores plotted in Fig. 8(a), (b), and (c) show that the `Jaccard` distance detects the best neighborhood function $N_{\mathcal{G}}^{\tau}$ for a threshold value of 0.8. As such, $N_{\mathcal{G}}^{\tau}$ is defined as follows:

$$N_{\mathcal{G}}^{\tau}(i) = \{x : \texttt{Jaccard}(x, i) \leq 0.8\}\,. \tag{31}$$

As expected, Geometric-AP achieves the best NMI value of 76.76% outperforming all other studied methods while the closest result has been attained by "Spectral-g" (76.01%). Meanwhile, for CR and F1, Geometric-AP remains comparable with the state-of-the-art graph clustering method "Spectral-g".

Obviously, the obtained results for recovering the various ground-truth class labels on the Zachary's karate club network have proven the consistent performance of our proposed method albeit the network information is redundant and explicitly extracted from the node features. This observation confirms that Geometric-AP effectively leverages the topological local neighborhood to better unveil irregularly shaped clusters associated with the analyzed data.

B. Clustering With Node Embeddings

Like the majority of machine learning algorithms, the performance of Geometric-AP greatly depends on data representation. For instance, different node embeddings may entangle or expose more or less the structure of the clusters present in the data [48]. With the fact that exemplar-based clustering methods are more successful with regularly shaped structures, representation learning becomes a key factor in achieving satisfactory clustering results. To provide a proof of concept that Geometric-AP can seamlessly be integrated with state-of-the-art representation learning methods while maintaining its efficiency we replace the node features used in the Zachary's Karate Club network by two different node embeddings obtained by two embedding methods widely used in the literature. We first consider the Fruchterman-Reingold force-directed algorithm (FRFD) that mimics forces in natural systems to embed undirected graphs in two dimensional spaces [49]. Similarly, we use t-SNE (t-distributed Stochastic Neighbor Embedding) method [50] that is well suited for visualization of high-dimensional datasets to project the network into the plane. Both embedding methods are observed as an aggregation of dimensionality reduction and representation learning techniques. As FRFD and t-SNE are randomly initialized, we generate 1000 sets of node embeddings per method and we analyze the average clustering performance of the different clustering algorithms being tested. For each

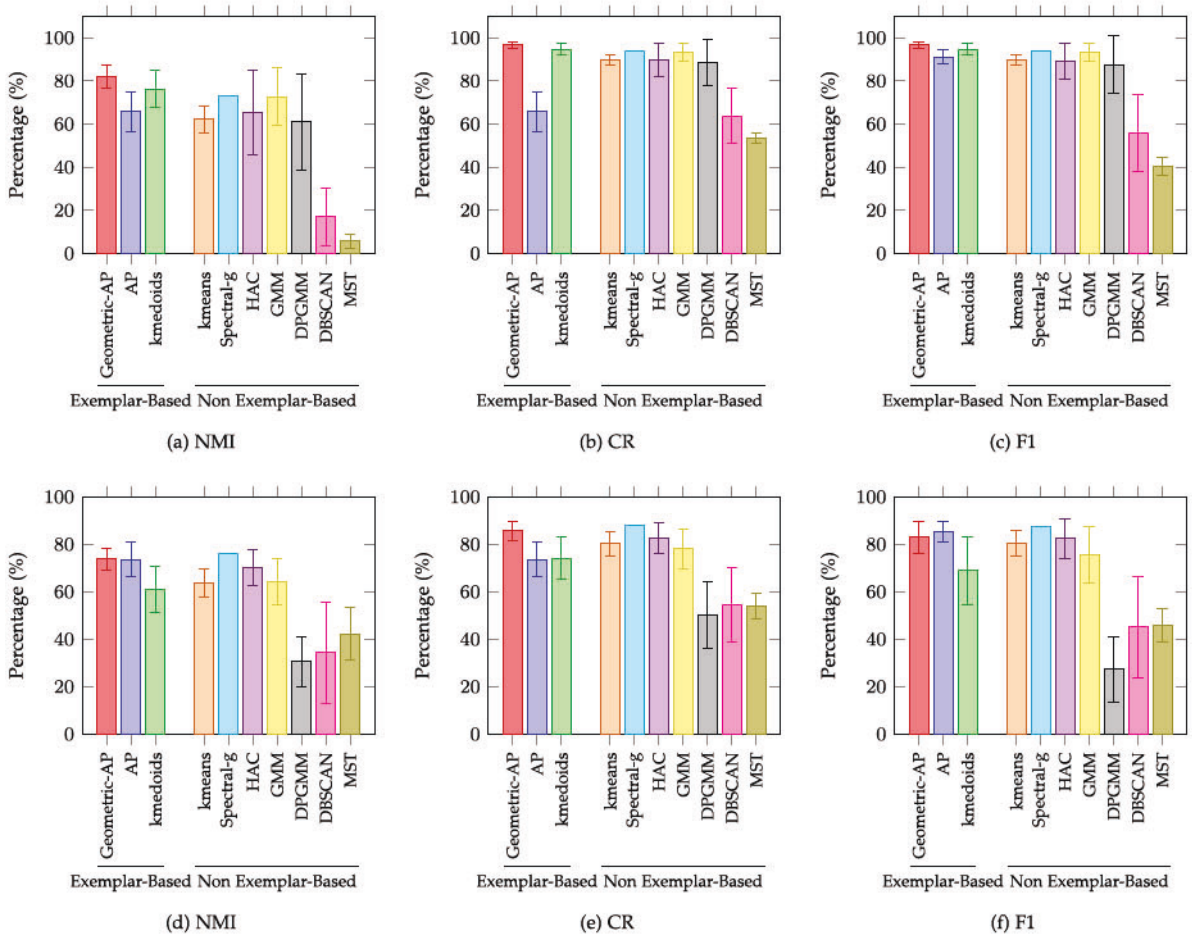


Fig. 9. Average clustering results using 1000 sets of FRFD node embeddings. Error bars show the standard deviations. (a-c) Results when using the club split classes. (d-f) Results when using the modularity-based classes.

evaluation metric we also report the standard deviation. For all sets of node embeddings we consider the negative Euclidean distance as the affinity measure for both Geometric-AP and AP. Hereafter, we tune the neighborhood function $N_{\mathcal{G}}^{\tau}$ for an arbitrary selected random seed and we retain the best configuration throughout all other repetitions. In our simulations we have used a random seed of value 13579. We also provide in Section II of the supplemental material, available online a visualization example of clustering by Geometric-AP that illustrates the efficacy of the proposed method in identifying high quality exemplars. Notice that for each set of node embeddings we produce the desired number of clusters by automatically adjusting the shared preference s_{ii} via dichotomic search. Node embeddings for which Geometric-AP or AP diverges are regenerated.

1) Fruchterman-Reingold Force-Directed Embeddings: We produce 1000 sets of FRFD node embeddings and we run Geometric-AP and other benchmark algorithms presented in Section IV-A for two clustering tasks. At first stage, we use the actual classes resulted from the club split. Then, we consider the classes obtained by modularity-based clustering. We report histograms of average NMI, CR, and F1 score values along with standard deviation bars for the tested algorithms. For methods that do not depend on node features such as Spectral-g or for those that show negligible standard deviations ($< 10^{-5}$) we omit the error bars.

Fig. 9(a), (b), and (c) show that a significant improvement has been achieved by Geometric-AP in modeling the correct club split. Additionally, Geometric-AP manifested the lowest variability on the three evaluation metrics. The reason is that the network information makes the algorithm less sensitive to the random noise associated with node features. Likewise, Fig. 9(d), (e), and (f) illustrate the consistent performance of Geometric-AP in retrieving the correct modularity-based classes. Overall, Geometric-AP remained the most robust method against the randomness associated with the used embeddings.

2) t-SNE Embeddings: Similarly, we employ the t-SNE algorithm to generate 1000 sets of node embeddings and we use the same setting discussed in Section V-B1 to run the simulations. We plot in Fig. 10(a), (b), (c) and 10(d), (e), and (f) the clustering results when using the club split classes and the modularity-based labels, respectively. Obviously, Geometric-AP outperforms exemplar-based methods by significant margins in all evaluation metrics. Also, it surpasses all feature-dependent methods including the state-of-the-art kmeans. On the other hand, Geometric-AP performs either comparably or

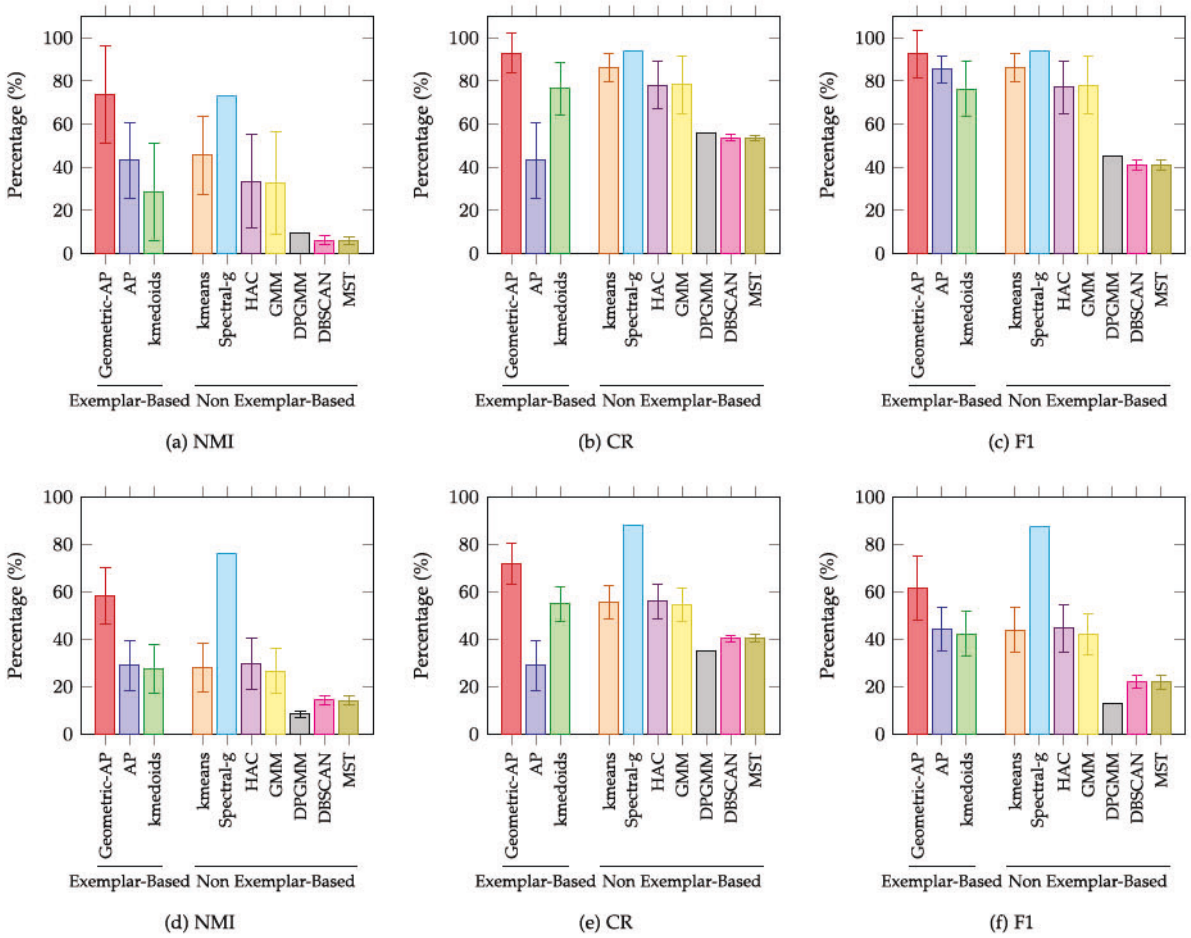


Fig. 10. Average clustering results using 1000 sets of t-SNE node embeddings. Error bars show the standard deviations. (**a-c**) Results when using the club split classes. (d-f) Results when using the modularity-based classes.

proximally to the popular graph clustering method Spectral-g even though Geometric-AP is not principally designed for graph clustering.

By considering the clustering performance of Geometric-AP using node embeddings, the comparative results establish the steady efficiency of Geometric-AP in leveraging the available network information to boost the clustering accuracy. Additionally, Geometric-AP demonstrates high compatibility with representation-learning-based methods and shows promising potentials if combined with data-driven methods.

VI. Insights and Limitations

Our analyses have shown the potential advantages of using the available network information in conjunction with data features to perform an efficient clustering. The performance of Geometric-AP on citation and social networks has demonstrated the ability of topological information to shrink the search space of the optimal clustering configuration without increasing the model complexity (see Table I). Theoretically, the accurate topological information leveraged by Geometric-AP helps to avoid some local optimum traps during the iterative search towards the optimal clustering. Such local optima are penalized by the topological constraint in the minimized energy function derived for Geometric-AP. Indeed, the hard constraint, set for distant exemplars with respect to the network, inhibits data points from selecting distant centroids in the presence of proximal ones. This selection procedure is consolidated by the new cluster assignment policy proposed by Geometric-AP that prioritizes closer neighbor exemplars in the network. As there is no guarantee that a given data point finds a good exemplar in its topological proximity, Geometric-AP accounts also for misassignments that could arise during the clustering. This is enabled by a label smoothing strategy that relies on graph coverings to determine more confidently the correct cluster labels.

On the other hand, achieving the best performance for Geometric-AP is contingent upon the identification of the appropriate topological distance metric and threshold to be used to characterize the neighborhood function N_G^T. In order to facilitate the utilization of our algorithm, we provide a few guidelines on how to tune the framework hyperparameters. Our analyses have shown that, based on the interactions between the cluster members and the characteristics of the network information, two classes of distance metrics can be considered. The first class corresponds to topological measures, such as the Cosine and Jaccard distances, that quantify the shared neighborhood between

TABLE I
TIME COMPLEXITY STATISTICS ON THE *CORA*, *CITESEER*, AND *ZACHARY'S KARATE CLUB* DATASETS IN TERMS OF EXECUTION TIME TO CONVERGENCE EXPRESSED IN SECONDS

	Cora								Citeseer								Zachary's Karate Club	
Nb Clusters	5	7	9	11	13	15	17	19	4	6	8	10	12	14	16	18	2	4
Geometric_AP	28.18	58.62	58.78	61.45	59.14	59.05	61.96	73.73	93.34	92	110.88	94.88	92.41	91.96	92.87	93.58	0.04	0.05
AP	**37.32**	28.41	31.67	54.63	26.4	52.14	54.8	59.19	87.27	77.22	79.34	85.22	86.24	87.15	88.45	89.74	0.05	0.05
Kmedoids	0.33	0.35	0.4	0.4	0.35	0.37	0.37	0.48	1.07	1.01	1.15	0.99	1.55	1.02	1	1.01	0.01	0.01
Kmeans	209.012	**271.83**	**339.04**	**413.21**	**450.43**	**520.75**	**600.44**	**723.57**	593.22	**780.19**	**1067.34**	**1203.65**	**1370.9**	**1607.96**	**1839.97**	**1968.41**	**1.4**	2.12
Spectral-g	11.2	13.28	19.18	18.8	20.66	23.39	28.48	28.59	15.72	23.03	40.14	41.6	44.38	44.44	49.08	54.9	1.32	**1.83**
HAC	7.46	7.37	7.76	7.27	7.43	7.39	7.51	7.63	28.85	28.86	30.16	28.72	28.37	28.89	28.26	28.39	0.01	0.02
GMM	**582.84**	846.77	1126.17	**1426.18**	**1526.56**	1731.56	2207.53	2424.63	**2899.3**	5735.92	9483.89	**11996.24**	14961.14	17440.43	18617.79	24440.8	**0.58**	0.9
DPGMM	666.5	**989.99**	**1271.62**	1565.4	1775.12	**2004**	**2540.56**	**2877.02**	**4921.63**	**6535.67**	**10409.62**	13431.46	**17365.51**	**19509.34**	**21727.06**	**28654.25**	**0.66**	0.1
DBSCAN	0.44	0.42	0.41	0.45	0.41	0.43	0.42	0.46	1.16	1.28	1.07	1.3	1.19	1.13	1.11	1.14	0.001	0.001
MST	2.41	2.62	2.52	2.58	2.4	2.38	2.53	2.95	4.87	4.31	4.24	4.4	4.71	4.36	4.33	4.35	0.03	0.003

Top three methods in terms of classification rate accuracy are expressed in bold font for each clustering scenario. top performing method with the minimum execution time is also underlined.

a pair of data points. Such distance metrics are well suited for small datasets (hundreds of data points) where the actual cluster members have direct interactions with respect to the network structure. This observation has been confirmed by our analysis on the *Zachary's Karate Club* dataset. Based on the desired clustering granularity we can also select the appropriate threshold τ. If the desired clustering is finely fragmented, it is more likely that the exemplar corresponding to any data point rests within a small radius from its neighborhood and setting a large threshold τ will better lead the search for such close exemplars (On the *Zachary's Karate Club* dataset, with modularity-based classes we found optimal $\tau = 0.8$). In contrast, coarse clustering requires a larger topological radius and hence a smaller τ (unlike to shortest path distance, we note that for Cosine and Jaccard distance metrics, τ is inversely proportional to the neighborhood radius). For instance, considering the club split classes, the best τ was equal to 0.5.

The second class of distance metrics probes the neighborhood depth within a network (i.e.,: shortest path distance). Such measures are adequate for large datasets (more than thousands of data points) where the good exemplars are within few hops w.r.t. the network from their cluster members (i.e.,: citation networks). To better select τ in this case, we can rely on the network density that reflects the degree of interactions between the network nodes. As the network becomes more sparse, τ needs to be increased to look for broader topological neighborhood. These recommendations are also consolidated by our findings on the *cora* and *citeseer* datasets. Nevertheless, in the absence of any prior domain knowledge about the clustering structure and the network properties, the parameter tuning can be exploratory based on the available data.

Despite the aforementioned advantages of Geometric-AP over traditional approaches, relying on network information can also have limitations. For instance, noisy or inaccurate network information could be misleading for the clustering task. For Geometric-AP, to be successful in dissecting the complex structure of data clusters requires the node features and the network topology to be consistent. In other words, the topology-based similarity between clusters members should not contradict their feature-based similarity. This consistency rule has been verified by our ablation experiments in Section III of the supplemental material, available online where the usage of a randomized network information has significantly deteriorated the performance of Geometric-AP. This setting presents a major challenge for our proposed approach to determine which information is more reliable.

From this perspective, a potential improvement of Geometric-AP is to account for the possible uncertainty related to the accuracy of the network information. For instance, one may quantify the differences between two clustering tasks carried with and without network information and set a decision threshold for using the topology information. A recently established approach introduces the identifiability [51] of communities as the discrepancy between the similarity matrix extracted from the node features and the normalized community incidence matrix extracted from the network information. A promising research direction is to leverage the identifiability of communities to quantify the improvement of the clustering after considering the network information. Another direction could be based on the latest advances in network vulnerability analysis [52] where the network information could be pruned based on Markov global connectivity metrics to keep only pertinent information. Alternatively, a transformation could be applied on the node features based on the available network information before conducting the clustering task. For example, an application for disease marker identification [53] has shown that relying on latent graph embeddings to cluster the gene nodes in a protein-protein interaction network (PPI) is very successful in discovering robust and reproducible biomarkers for complex disorders. Ultimately, an interesting research direction is to extend Geometric-AP by performing a linear fusion at the feature level for two similarity matrices that encode for the node features and the network topology. As performed in [54], deriving an estimate for the latent similarities by optimizing the cosine similarity between the estimate and the latent representations could be very efficient in leveraging the network information.

VII. CONCLUSION

In this article, we proposed a novel geometric clustering scheme, Geometric-AP, by extending the original feature-based AP to effectively take advantage of network relations between the data points, often available in various scientific datasets. Geometric-AP locks its focus on the local network neighborhood during the message updates and makes potential exemplars only available within a predefined topological sphere in the

network. The underlying objective is to maximize the similarity between the data points and their respective exemplars based on the node features while respecting the network topology to ensure the proximity of each data point from its exemplar in the network. Using max-sum belief propagation over a factor graph, the new model has been optimized under the given network constraints at the level of function nodes. With an adjusted cluster assignment policy, the hybrid model further smooths the node labels throughout the network via majority voting. By initially considering all data points as potential exemplars, Geometric-AP generates clusters insensitive to initialization.

Extensive validation based on two benchmark citation networks and one social network has clearly demonstrated the effectiveness of the proposed method and has confirmed the statistical significance of the obtained results. It has been shown that Geometric-AP results in higher accuracy and robustness than the original AP in clustering the data points by leveraging relevant network knowledge. Furthermore, comparative performance assessment against other state-of-the-art methods have shown that Geometric-AP consistently yields favorable clustering results.

REFERENCES

[1] A. K. Jain, N. Murty, and P. Flynn, "Data clustering: A review," *ACM Comput. Surveys*, vol. 31, no. 3, pp. 264–323, Sep. 1999.

[2] L. Danial and G. Polina, "Convex clustering with exemplar-based models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 825–832.

[3] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A kmeans clustering algorithm," *J. Roy. Statist. Soc.*, vol. 28, no. 1, pp. 100–108, 1979.

[4] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis* (Wiley Series in Probability and Statistics). Hoboken, New Jersey, USA: John Wiley, 1990. [Online]. Available: https://doi.org/10.1002/9780470316801

[5] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. 14th Int. Conf. Neural Inf. Process. Syst.*, 2001, pp. 849–856.

[6] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 4412–4419.

[7] W. Zhuge, H. Tao, T. Luo, L.-L. Zeng, C. Hou, and D. Yi, "Joint representation learning and clustering: A framework for grouping partial multiview data," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 8, pp. 3826–3840, Aug. 2022.

[8] H. Tao, C. Hou, Y. Qian, J. Zhu, and D. Yi, "Latent complete row space recovery for multi-view subspace clustering," *IEEE Trans. Image Process.*, vol. 29, pp. 8083–8096, 2020.

[9] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2014, pp. 701–710.

[10] N. T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th Int. Conf. Learn. Representations*, 2017.

[11] X. Zhang, H. Liu, Q. Li, and X.-M. Wu, "Attributed graph clustering via adaptive graph convolution," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 376–381.

[12] F. E. Maranzana, "On the location of supply points to minimize transport costs," *J. Oper. Res. Soc.*, vol. 15, no. 3, pp. 261–270, 1964.

[13] B. Heller, R. Sherwood, and N. McKeown, "The controller placement problem," in *Proc. 1st Workshop Hot Topics Softw. Defined Netw.*, 2012, pp. 7–12.

[14] H.-J. Li, Z. Bu, Z. Wang, and J. Cao, "Dynamical clustering in electronic commerce systems via optimization and leadership expansion," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5327–5334, Aug. 2020.

[15] J. B. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976,2007.

[16] S. Parthasarathy, R. J. Yoder, and V. Satuluri, "Community discovery in social networks: Applications, methods and emerging trends," *Social Network Data Analytics*, Berlin, Germany: Springer, 2011, pp. 79–113.

[17] K. Mitra, A.-R. Carvunis, S. K. Ramesh, and T. Ideker, "Integrative approaches for finding modular structure in biological networks," *Nature Rev. Genet.*, vol. 14, no. 10, pp. 719–732, 2013.

[18] Y. Wang, H. Jeong, B.-J. Yoon, and X. Qian, "ClusterM: A scalable algorithm for computational prediction of conserved protein complexes across multiple protein interaction networks," *BMC Genomic.*, vol. 21, 2020, Art. no. 615.

[19] X. Zhang, H. Su, and H.-H. Chen, "Cluster-based multi-channel communications protocols in vehicle ad hoc networks," *IEEE Wireless Commun.*, vol. 13, no. 5, pp. 44–51, May 2006.

[20] H.-J. Li, W. Xu, S. Song, W.-X. Wang, and M. Perc, "The dynamics of epidemic spreading on signed networks," *Chaos, Solitons Fractals*, vol. 151, 2021, Art. no. 111294.

[21] M. Leone and S. Martin Weigt, "Clustering by soft-constraint affinity propagation: Applications to gene-expression data," *Bioinformatics*, vol. 23, no. 20, pp. 2708–2715, Oct. 2007.

[22] J. Xiao, J. Wang, P. Tan, and L. Quan, "Joint affinity propagation for multiple view segmentation," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, 2007, pp. 1–7.

[23] E. Inmar, C. G. Chung, and B. J. Frey, "Hierarchical affinity propagation," in *Proc. 27th Conf. Uncertainty Artif. Intell.*, 2011, pp. 238–246.

[24] E. I. Givoni and B. J. Frey, "Semi-supervised affinity propagation with instance-level constraints," in *Proc. 12th Int. Conf. Artif. Intell. Statist.*, 2009, pp. 161–168.

[25] N. M. Arzeno and H. Vikalo, "Semi-supervised affinity propagation with soft instance-level constraints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 5, pp. 1041–1052, May 2014.

[26] X. Zhang, C. Furtlehner, and M. Sebag, "Data streaming with affinity propagation," *Mach. Learn. Knowl. Discov. Databases*, vol. 5212, pp. 628–643, 2008.

[27] D. Tarlow, R. S. Zemel, and B. J. Frey, "Flexible priors for exemplar-based clustering," in *Proc. 24th Conf. Uncertainty Artif. Intell.*, 2008, pp. 537–545.

[28] C.-D. Wang, J.-H. Lai, C. Y. Suen, and J.-Y. Zhu, "Multi-exemplar affinity propagation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2223–2237, Sep. 2013.

[29] H.-W. Liu, "Community detection by affinity propagation with various similarity measures," in *Proc. IEEE Int. Joint Conf. Comput. Sci. Optim.*, 2011, pp. 182–186.

[30] S. Taheri and A. Bouyer, "Community detection in social networks using affinity propagation with adaptive similarity matrix," *Big Data*, vol. 8, no. 3, pp. 189–202, 2020.

[31] E. S. Schaeffer, "Graph clustering," *Comput. Sci. Rev.*, vol. 1, no. 1, pp. 27–64, 2007.

[32] A. Guillory and J. Bilmes, "Label selection on graphs," in *Proc. 22nd Int. Conf. Neural Inf. Process. Syst.*, 2009, pp. 691–699.

[33] A. K. McCallum, K. Nigam, J. Rennie, and K. Seymore, "Automating the construction of internet portals with machine learning," *Inf. Retrieval J.*, vol. 3, no. 2, pp. 127–163, 2000.

[34] C. L. Giles, K. D. Bollacker, and S. Lawrence, "CiteSeer: An automatic citation indexing system," in *Proc. 3rd ACM Conf. Digit. Libraries*, 1998, pp. 89–98.

[35] D. Dueck, "Affinity propagation: Clustering data by passing messages," PhD dissertation, University of Toronto, Toronto, Canada, 2009.

[36] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[37] J. H. Ward, "Hierarchical grouping to optimize an objective function," *J. Amer. Statist. Assoc.*, vol. 58, no. 301, pp. 236–244, 1963.

[38] G. J. McLachlan and K. E. Basford, "Mixture models: Inference and applications to clustering," *Statist. Textbooks Monographs Marcel Dekker*, vol. 84, pp. 1–254, 1988.

[39] D. M. Blei and M. I. Jordan, "Variational inference for dirichlet process mixtures," *Bayesian Anal.*, vol. 1, no. 1, pp. 121–143, 2006.

[40] S. M. Wong and Y. Y. Yao, "An information theoretic measure of term specificity," *J. Amer. Soc. Inf. Sci.*, vol. 43, no. 1, pp. 54–61, 1992.

[41] A. Strehl, J. Ghosh, and R. Mooney, "Impact of similarity measures on web-page clustering," in *Proc. Workshop Artif. Intell. Web Search*, 2000, pp. 58–64.

[42] C. Charu Aggarwal and C. K. Reddy, *Data Clustering: Algorithms and Applications*. London, U.K.: Chapman & Hall/CRC, 2013.

[43] C. Yang, Z. Liu, D. Zhao, M. Sun, and E. Y. Chang, "Network representation learning with rich text information," in *Proc. 24th Int. Conf. Artif. Intell.*, 2015, pp. 2111–2117.

[44] A. R. Rossi and N. K. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 4292–4293.

[45] W. W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropological Res.*, vol. 33, no. 4, pp. 452–473, 1977.
[46] U. Brandes et al., "On modularity clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 2, pp. 172–188, Feb. 2008.
[47] E. J. M. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Review. E. Statist. Nonlinear Soft Matter Phys.*, vol. 69, no. 2, 2004, Art. no. 026113.
[48] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[49] M. J. T. Fruchterman and E. M. Reingold, "Graph drawing by force-directed placement," *Softw.: Pract. Experience*, vol. 21, pp. 1129–1164, 1991.
[50] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.
[51] H.-J. Li, L. Wang, Y. Zhang, and Matjaž Perc, "Optimization of identifiability for efficient community detection," *New J. Phys.*, vol. 22, 2020, Art. no. 063035.
[52] H.-J. Li, L. Wang, Z. Bu, J. Cao, and Y. Shi, "Measuring the network vulnerability based on markov criticality," *ACM Trans. Knowl. Discov. Data*, vol. 16, no. 2, pp. 1–24, 2021.
[53] O. Maddouri, X. Qian, and B.-J. Yoon, "Deep graph representations embed network information for robust disease marker identification," *Bioinformatics*, vol. 38, no. 4, pp. 1075–1086, 2022.
[54] H.-J. Li, Z. Wang, J. Cao, J. Pei, and Y. Shi, "Optimal estimation of low-rank factors via feature level data fusion of multiplex signal systems," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 6, pp. 2860–2871, Jun. 2022.

Omar Maddouri received the BS degree in computer science from the National School of Computer Science (ENSI), Manouba, Tunisia, in 2010, the MS degree in biological and biomedical sciences from Hamad Bin Khalifa University (HBKU), Doha, Qatar, in 2017, and the MS and PhD degrees in electrical engineering from Texas A&M University, College Station, TX, USA, in 2019 and 2022, respectively. Between 2010 and 2015, he served as a software engineer in LG Electronics and Continental Automotive. He is currently an Artificial Intelligence and Machine Learning Scientist at Ford Motor Company, Dearborn, MI, USA. His main research interests include Bayesian inference, transfer learning, structured data analysis, representation learning, and active learning.

Xiaoning Qian (Senior Member, IEEE) received the BSE degree from Shanghai Jiao-Tong University, and the MS, MPh, and PhD degrees in electrical engineering from Yale University, New Haven, CT, USA. He is currently a professor with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. He also holds a joint appointment with Brookhaven National Laboratory, Upton, NY, where he is a scientist in Computational Science Initiative, Applied Mathematics group. His recent honors include the National Science Foundation CAREER Award, the Texas A&M Engineering Experiment Station (TEES) Faculty Fellow, and the Montague-Center for Teaching Excellence Scholar with Texas A&M University. His research interests include machine learning, Bayesian computation, and their applications in materials science, computational network biology, genomic signal processing, and biomedical signal and image analysis.

Byung-Jun Yoon (Senior Member, IEEE) received the BSE (summa cum laude) degree from Seoul National University, Seoul, South Korea, in 1998, and the MS and PhD degrees from the California Institute of Technology, Pasadena, CA, USA, in 2002 and 2007, respectively, all in electrical engineering. He joined the Department of Electrical and Computer Engineering, Texas A&M University, College Station, in 2008, where he is currently an associate professor. Dr. Yoon also holds a joint appointment at Brookhaven National Laboratory, Upton, NY, where he is a Scientist in Computational Science Initiative, Applied Mathematics group. His honors include the National Science Foundation CAREER Award, the Best Paper Award at the 9th Asia Pacific Bioinformatics Conference, the Best Paper Award with the 12th Annual MCBIOS Conference, and the SLATE Teaching Excellence Award from the Texas A&M University System. His main research interests include optimal experimental design, objective-based uncertainty quantification, and scientific machine learning for a wide range of applications, including bioinformatics, computational network biology, materials science, and drug discovery.