An Approximation Algorithm for a Semi-supervised Graph Clustering Problem

Victor Il'ev[1,2(✉)], Svetlana Il'eva[1], and Alexander Morshinin[2]

[1] Dostoevsky Omsk State University, Omsk, Russia
iljev@mail.ru
[2] Sobolev Institute of Mathematics SB RAS, Omsk, Russia
morshinin.alexander@gmail.com

Abstract. Clustering problems form an important section of data analysis. In machine learning clustering problems are usually classified as unsupervised learning. Semi-supervised clustering problems are also considered. In these problems relatively few objects are labeled (i.e., are assigned to clusters), whereas a large number of objects are unlabeled.

We consider the most visual formalization of a version of semi-supervised clustering. In this problem one has to partition a given set of n objects into k clusters ($k < n$). A collection of k pairwise disjoint nonempty subsets of objects is fixed. No two objects from different subsets of this collection may belong to the same cluster and all objects from any subset must belong to the same cluster. Similarity of objects is determined by an undirected graph. Vertices of this graph are in one-to-one correspondence with objects, and edges connect similar objects. One has to partition the vertices of the graph into pairwise disjoint groups (clusters) minimizing the number of edges between clusters and the number of missing edges inside clusters.

The problem is NP-hard for any fixed $k \geq 2$. For $k = 2$ we present a polynomial time approximation algorithm and prove a performance guarantee of this algorithm.

Keywords: Graph clustering · Approximation algorithm · Performance guarantee

1 Introduction

The objective of clustering problems is to partition a given set of objects into a family of subsets (called *clusters*) such that objects within a cluster are more similar to each other than objects from different clusters. In pattern recognition and machine learning clustering methods fall under the section of *unsupervised learning*. At the same time, *semi-supervised* clustering problems are studied. In these problems relatively few objects are labeled (i.e., are assigned to clusters), whereas a large number of objects are unlabeled [1,3].

Y. Kochetov et al. (Eds.): MOTOR 2020, CCIS 1275, pp. 23–29, 2020.
https://doi.org/10.1007/978-3-030-58657-7_3

One of the most visual formalizations of clustering is the *graph clustering*, that is, grouping the vertices of a graph into clusters taking into consideration the edge structure of the graph. In this paper, we consider three interconnected versions of graph clustering, two of which are semi-supervised ones.

We consider only *simple* graphs, i.e., undirected graphs without loops and multiple edges. A graph is called a *cluster graph*, if each of its connected components is a complete graph [6].

Let V be a finite set. Denote by $\mathcal{M}(V)$ the set of all cluster graphs on the vertex set V. Let $\mathcal{M}_k(V)$ be the set of all cluster graphs on V consisting of exactly k nonempty connected components, $2 \leq k \leq |V|$.

If $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ are graphs on the same labeled vertex set V, then the *distance* $\rho(G_1, G_2)$ between them is defined as follows

$$\rho(G_1, G_2) = |E_1 \Delta E_2| = |E_1 \setminus E_2| + |E_2 \setminus E_1|,$$

i.e., $\rho(G_1, G_2)$ is the number of noncoinciding edges in G_1 and G_2.

Consider three interconnected graph clustering problems.

GC$_k$ (Graph k-Clustering). Given a graph $G = (V, E)$ and an integer k, $2 \leq k \leq |V|$, find a graph $M^* \in \mathcal{M}_k(V)$ such that

$$\rho(G, M^*) = \min_{M \in \mathcal{M}_k(V)} \rho(G, M).$$

SGC$_k$ (Semi-supervised Graph k-Clustering). Given a graph $G = (V, E)$, an integer k, $2 \leq k \leq |V|$, and a set $Z = \{z_1, \dots z_k\} \subset V$ of pairwise different vertices, find $M^* \in \mathcal{M}_k(V)$ such that

$$\rho(G, M^*) = \min_{M \in \mathcal{M}_k(V)} \rho(G, M),$$

where minimum is taken over all cluster graphs $M = (V, E_M) \in \mathcal{M}_k(V)$ with $z_i z_j \notin E_M$ for all $i, j \in \{1, \dots k\}$ (in other words, all vertices of Z belong to different connected components of M).

SSGC$_k$ (Set Semi-supervised Graph k-Clustering). Given a graph $G = (V, E)$, an integer k, $2 \leq k \leq |V|$, and a collection $\mathcal{Z} = \{Z_1, \dots Z_k\}$ of pairwise disjoint nonempty subsets of V, find $M^* \in \mathcal{M}_k(V)$ such that

$$\rho(G, M^*) = \min_{M \in \mathcal{M}_k(V)} \rho(G, M),$$

where minimum is taken over all cluster graphs $M = (V, E_M) \in \mathcal{M}_k(V)$ such that

1. $zz' \notin E_M$ for all $z \in Z_i, z' \in Z_j$, $i, j = 1, \dots, k$, $i \neq j$;
2. $zz' \in E_M$ for all $z, z' \in Z_i$, $i = 1, \dots, k$

(in other words, all sets of the family $\mathcal{Z}$ are subsets of different connected components of M).

Problem $\mathbf{GC}_k$ is NP-hard for every fixed $k \geq 2$ [6]. It is not difficult to construct Turing reduction of problem $\mathbf{GC}_k$ to problem $\mathbf{SGC}_k$ and as a result to show that $\mathbf{SGC}_k$ is NP-hard too. Thus, problem $\mathbf{SSGC}_k$ is also NP-hard as generalization of $\mathbf{SGC}_k$.

In 2004, Bansal, Blum, and Chawla [2] presented a polynomial time 3-approximation algorithm for a version of the graph clustering problem similar to $\mathbf{GC}_2$ in which the number of clusters doesn't exceed 2. In 2008, Coleman, Saunderson, and Wirth [4] presented a 2-approximation algorithm for this version applying local search to every feasible solution obtained by the 3-approximation algorithm from [2]. They used a switching technique that allows to reduce clustering any graph to the equivalent problem whose optimal solution is the complete graph, i.e., the cluster graph consisting of the single cluster. In [5], we presented a modified 2-approximation algorithm for problem $\mathbf{GC}_2$. In contrast to the proof of Coleman, Saunderson, and Wirth, our proof of the performance guarantee of this algorithm didn't use switchings.

In this paper, we use a similar approach to construct a 2-approximation local search algorithm for the set semi-supervised graph clustering problem $\mathbf{SSGC}_2$. Applying this method to problem $\mathbf{SGC}_2$ we get a variant of 2-approximation algorithm for this problem.

2 Problem $\mathbf{SSGC}_2$

2.1 Notation and Auxiliary Propositions

Consider the special case of problem $\mathbf{SSGC}_k$ with $k = 2$. We need to introduce the following notation.

Given a graph $G = (V, E)$ and a vertex $v \in V$, we denote by $N_G(v)$ the set of all vertices adjacent to v in G, and let $\overline{N}_G(v) = V \setminus (N_G(v) \cup \{v\})$.

Let $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ be graphs on the same labeled vertex set V, $n = |V|$. Denote by $D(G_1, G_2)$ the graph on the vertex set V with the edge set $E_1 \Delta E_2$. Note that $\rho(G_1, G_2)$ is equal to the number of edges in the graph $D(G_1, G_2)$.

Lemma 1. *[5] Let $d_{\min}$ be the minimum vertex degree in the graph $D(G_1, G_2)$. Then*

$$\rho(G_1, G_2) \geq \frac{nd_{\min}}{2}.$$

Let $G = (V, E)$ be an arbitrary graph. For any vertex $v \in V$ and a set $A \subseteq V$ we denote by A_v^+ the number of vertices $u \in A$ such that $vu \in E$, and by A_v^- the number of vertices $u \in A \setminus \{v\}$ such that $vu \notin E$.

For nonempty sets $X, Y \subseteq V$ such that $X \cap Y = \emptyset$ and $X \cup Y = V$ we denote by $M(X, Y)$ the cluster graph in $\mathcal{M}_2(V)$ with connected components induced by X, Y. The sets X and Y will be called *clusters*.

The following lemma was proved in [5] for problem $\mathbf{GC_2}$. Its proof for problem $\mathbf{SSGC_2}$ is exactly the same.

Lemma 2. *Let $G = (V, E)$ be an arbitrary graph, $M^* = M(X^*, Y^*)$ be an optimal solution to problem $\mathbf{SSGC_2}$ on the graph G, and $M = M(X, Y)$ be an arbitrary feasible solution to problem $\mathbf{SSGC_2}$ on the graph G. Then*

$$\rho(G, M) - \rho(G, M^*) =$$
$$\sum_{u \in X \cap Y^*} \Big((X \cap X^*)_u^- - (X \cap X^*)_u^+ + (Y \cap Y^*)_u^+ - (Y \cap Y^*)_u^- \Big) +$$
$$\sum_{u \in Y \cap X^*} \Big((Y \cap Y^*)_u^- - (Y \cap Y^*)_u^+ + (X \cap X^*)_u^+ - (X \cap X^*)_u^- \Big).$$

2.2 Local Search Procedure

Let us introduce the following local search procedure.

Procedure LS(M, X, Y, Z_1, Z_2).
Input: cluster graph $M = M(X, Y) \in \mathcal{M}_2(V)$, Z_1, Z_2 are disjoint nonempty sets, $Z_1 \subset X, Z_2 \subset Y$.
Output: cluster graph $L = M(X', Y') \in \mathcal{M}_2(V)$ such that $Z_1 \subseteq X'$, $Z_2 \subseteq Y'$.

Iteration 0. Set $X_0 = X, Y_0 = Y$.
Iteration k($k \geq 1$).
Step 1. For each vertex $u \in V \setminus (Z_1 \cup Z_2)$ calculate the following quantity $\delta_k(u)$ (possible variation of the value of the objective function in case of moving the vertex u to another cluster):

$$\delta_k(u) = \begin{cases} (X_{k-1})_u^- - (X_{k-1})_u^+ + (Y_{k-1})_u^+ - (Y_{k-1})_u^- & \text{for } u \in X_{k-1} \setminus Z_1, \\ (Y_{k-1})_u^- - (Y_{k-1})_u^+ + (X_{k-1})_u^+ - (X_{k-1})_u^- & \text{for } u \in Y_{k-1} \setminus Z_2. \end{cases}$$

Step 2. Choose the vertex $u_k \in V \setminus (Z_1 \cup Z_2)$ such that

$$\delta_k(u_k) = \max_{u \in V \setminus (Z_1 \cup Z_2)} \delta_k(u).$$

Step 3. If $\delta_k(u_k) > 0$, then set $X_k = X_{k-1} \setminus \{u_k\}$, $Y_k = Y_{k-1} \cup \{u_k\}$ in case of $u_k \in X_{k-1}$, and set $X_k = X_{k-1} \cup \{u_k\}$, $Y_k = Y_{k-1} \setminus \{u_k\}$ in case of $u_k \in Y_{k-1}$; **go to iteration $k + 1$**. Else **STOP**. Set $X' = X_{k-1}$, $Y' = Y_{k-1}$, and $L = M(X', Y')$.
End.

2.3 2-Approximation Algorithm for Problem $\mathbf{SSGC_2}$

Consider the following approximation algorithm for problem $\mathbf{SSGC_2}$.

Algorithm $\mathbf{A}_1$.
Input: graph $G = (V, E)$, Z_1, Z_2 are disjoint nonempty subsets of V.
Output: graph $M_1 = M(X, Y) \in \mathcal{M}_2(V)$, sets Z_1, Z_2 are subsets of different clusters.

Step 1. For every vertex $u \in V$ do the following:

Step 1.1. (a) If $u \notin Z_1 \cup Z_2$, then define the cluster graphs $\overline{M}_u = M(\overline{X}, \overline{Y})$ and $\overline{\overline{M}}_u = M(\overline{\overline{X}}, \overline{\overline{Y}})$, where

$$\overline{X} = \{u\} \cup \big((N_G(u) \cup Z_1) \setminus Z_2\big), \overline{Y} = V \setminus \overline{X},$$
$$\overline{\overline{X}} = \{u\} \cup \big((N_G(u) \cup Z_2) \setminus Z_1\big), \overline{\overline{Y}} = V \setminus \overline{\overline{X}}.$$

(b) If $u \in Z_1 \cup Z_2$, then define the cluster graph $M_u = M(X, Y)$, where

$$X = \{u\} \cup ((N_G(u) \cup Z) \setminus \overline{Z}), Y = V \setminus X.$$

Here $Z = Z_1$, $\overline{Z} = Z_2$ in case of $u \in Z_1$, and $Z = Z_2$, $\overline{Z} = Z_1$, otherwise.

Step 1.2. (a) If $u \notin Z_1 \cup Z_2$, then run the local search procedure $\mathbf{LS}(\overline{M}_u, \overline{X}, \overline{Y}, Z_1, Z_2)$ and $\mathbf{LS}(\overline{\overline{M}}_u, \overline{\overline{X}}, \overline{\overline{Y}}, Z_1, Z_2)$. Denote resulting graphs by $\overline{L}_u$ and $\overline{\overline{L}}_u$.

(b) If $u \in Z_1 \cup Z_2$, then run the local search procedure $\mathbf{LS}(M_u, X, Y, Z_1, Z_2)$. Denote resulting graph by L_u.

Step 2. Among all locally-optimal solutions L_u, $\overline{L}_u$, $\overline{\overline{L}}_u$ obtained at step 1.2 choose the nearest to G cluster graph $M_1 = M(X, Y)$.

The following lemma can be proved in the same manner as Remark 1 in [5].

Lemma 3. *Let $G = (V, E)$ be an arbitrary graph, Z_1, Z_2 be arbitrary disjoint nonempty subsets of V, $M^* = M(X^*, Y^*) \in \mathcal{M}_2(V)$ be an optimal solution to problem $\mathbf{SSGC_2}$ on the graph G, and $d_{\min}$ be the minimum vertex degree in the graph $D = D(G, M^*)$. Among all graphs M_u, $\overline{M}_u$, $\overline{\overline{M}}_u$ constructed by algorithm $\mathbf{A_1}$ at step 1.1 there is the cluster graph $M = M(X, Y)$ such that*

1. *M can be obtained from M^* by moving at most $d_{\min}$ vertices to another cluster;*
2. *If $Z_1 \subset X^*, Z_2 \subset Y^*$, then $Z_1 \subset X \cap X^*, Z_2 \subset Y \cap Y^*$. Otherwise, if $Z_2 \subset X^*, Z_1 \subset Y^*$, then $Z_1 \subset Y \cap Y^*, Z_2 \subset X \cap X^*$.*

Now we can prove a performance guarantee of algorithm $\mathbf{A_1}$.

Theorem 1. *For every graph $G = (V, E)$ and for any disjoint nonempty subsets $Z_1, Z_2 \subset V$ the following inequality holds:*

$$\rho(G, M_1) \le 2\rho(G, M^*),$$

where $M^* \in \mathcal{M}_2(V)$ is an optimal solution to problem $\mathbf{SSGC_2}$ on the graph G and $M_1 \in \mathcal{M}_2(V)$ is the solution returned by algorithm $\mathbf{A_1}$.

Proof. Let $M^* = M(X^*, Y^*)$ and $d_{\min}$ be the minimum vertex degree in the graph $D = D(G, M^*)$. By Lemma 3, among all graphs constructed by algorithm $\mathbf{A_1}$ at step 1.1 there is the cluster graph $M = M(X, Y)$ satisfying the conditions 1 and 2 of Lemma 3. By condition 1, $|X \cap Y^*| \cup |Y \cap X^*| \le d_{\min}$.

Consider the performance of procedure $\mathbf{LS}(M, X, Y, Z_1, Z_2)$ on the graph $M = M(X, Y)$.

Local search procedure **LS** starts with $X_0 = X$ and $Y_0 = Y$. At every iteration k either **LS** moves some vertex $u_k \in V \setminus (Z_1 \cup Z_2)$ to another cluster, or no vertex is moved and **LS** finishes.

Consider in detail iteration $t+1$ such that

– at every iteration $k = 1, \ldots, t$ procedure **LS** selects some vertex

$$u_k \in (X \cap Y^*) \cup (Y \cap X^*);$$

– at iteration $t+1$ either procedure **LS** selects some vertex

$$u_{t+1} \in \big((X \cap X^*) \cup (Y \cap Y^*)\big) \setminus (Z_1 \cup Z_2),$$

or iteration $t+1$ is the last iteration of **LS**.

Let us introduce the following quantities:

$$\alpha_{t+1}(u) = \begin{cases} (X_t \cap X^*)_u^- - (X_t \cap X^*)_u^+ + (Y_t \cap Y^*)_u^+ - (Y_t \cap Y^*)_u^- & \text{for } u \in X_t \cap Y^* \\ (Y_t \cap Y^*)_u^- - (Y_t \cap Y^*)_u^+ + (X_t \cap X^*)_u^+ - (X_t \cap X^*)_u^- & \text{for } u \in Y_t \cap X^*. \end{cases}$$

Consider the cluster graph $M_t = M(X_t, Y_t)$. By Lemma 2,

$$\rho(G, M_t) - \rho(G, M^*) = \sum_{u \in X_t \cap Y^*} \alpha_{t+1}(u) + \sum_{u \in Y_t \cap X^*} \alpha_{t+1}(u).$$

Put $r = |X_t \cap Y^*| + |Y_t \cap X^*|$. Since at all iterations preceding iteration $t+1$ only vertices from the set $(X \cap Y^*) \cup (Y \cap X^*)$ were moved, then

$$r = |X_t \cap Y^*| + |Y_t \cap X^*| \le d_{\min}. \tag{1}$$

Hence

$$\rho(G, M_t) - \rho(G, M^*) \le r \max\{\alpha_{t+1}(u) : u \in (X_t \cap Y^*) \cup (Y_t \cap X^*)\}. \tag{2}$$

Note that at iteration $t+1$ for every vertex $u \in (X_t \cap Y^*) \cup (Y_t \cap X^*)$ the following inequality holds:

$$\alpha_{t+1}(u) \le \frac{n}{2}. \tag{3}$$

The proof of this inequality is similar to the proof of inequality (5) in [5].

Denote by L the graph returned by procedure $\mathbf{LS}(M, X, Y, Z_1, Z_2)$. Using (1), (2), (3), and Lemma 1 we obtain

$$\rho(G, L) - \rho(G, M^*) \le \rho(G, M_t) - \rho(G, M^*) \le$$
$$r \max\{\alpha_{t+1}(u) : u \in (X_t \cap Y^*) \cup (Y_t \cap X^*)\} \le r\frac{n}{2} \le d_{\min}\frac{n}{2} \le \rho(G, M^*).$$

Thus, $\rho(G, L) \leq 2\rho(G, M^*)$.

The graph L is constructed among all graphs L_u, $\overline{L}_u$, $\overline{\overline{L}}_u$ at step 1.2 of algorithm $\mathbf{A_1}$. Performance guarantee of algorithm $\mathbf{A_1}$ follows.

Theorem 1 is proved.

It is easy to see that problem $\mathbf{SGC}_2$ is a special case of problem $\mathbf{SSGC}_2$ if $|Z_1| = |Z_2| = 1$. The following theorem is the direct corollary of Theorem 1.

Theorem 2. *For every graph $G = (V, E)$ and for any subset $Z = \{z_1, z_2\} \subset V$ the following inequality holds:*

$$\rho(G, M_1) \leq 2\rho(G, M^*),$$

where $M^* \in \mathcal{M}_2(V)$ is an optimal solution to problem $\mathbf{SGC_2}$ on the graph G and $M_1 \in \mathcal{M}_2(V)$ is the solution returned by algorithm $\mathbf{A_1}$.

References

1. Bair, E.: Semi-supervised clustering methods. Wiley Interdisc. Rev. Comput. Stat. **5**(5), 349–361 (2013)
2. Bansal, N., Blum, A., Chawla, S.: Correlation clustering. Mach. Learn. **56**, 89–113 (2004)
3. Chapelle, O., Schölkopf, B., Zein, A.: Semi-Supervised Learning. MIT Press, Cambridge (2006)
4. Coleman, T., Saunderson, J., Wirth, A.: A local-search 2-approximation for 2-correlation-clustering. In: Halperin, D., Mehlhorn, K. (eds.) ESA 2008. LNCS, vol. 5193, pp. 308–319. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-87744-8_26
5. Il'ev, V., Il'eva, S., Morshinin, A.: A 2-approximation algorithm for the graph 2-clustering problem. In: Khachay, M., Kochetov, Y., Pardalos, P. (eds.) MOTOR 2019. LNCS, vol. 11548, pp. 295–308. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-22629-9_21
6. Shamir, R., Sharan, R., Tsur, D.: Cluster graph modification problems. Discrete Appl. Math. **144**(1–2), 173–182 (2004)