Semi-Supervised Clustering via Information-Theoretic Markov Chain Aggregation

Sophie Steger
Signal Processing and Speech Communication Laboratory
Graz University of Technology
Graz, Austria
sophie.steger@student.tugraz.at

Bernhard C. Geiger
Know-Center GmbH
Graz, Austria
geiger@ieee.org

Marek Śmieja
Faculty of Mathematics and Computer Science
Jagiellonian University
Krakow, Poland
marek.smieja@uj.edu.pl

ABSTRACT

We connect the problem of semi-supervised clustering to constrained Markov aggregation, i.e., the task of partitioning the state space of a Markov chain. We achieve this connection by considering every data point in the dataset as an element of the Markov chain's state space, by defining the transition probabilities between states via similarities between corresponding data points, and by incorporating semi-supervision information as hard constraints in a Hartigan-style algorithm. The introduced Constrained Markov Clustering (CoMaC) is an extension of a recent information-theoretic framework for (unsupervised) Markov aggregation to the semi-supervised case. Instantiating CoMaC for certain parameter settings further generalizes two previous information-theoretic objectives for unsupervised clustering. Our results indicate that CoMaC is competitive with the state-of-the-art.

CCS CONCEPTS

• **Computing methodologies** → **Semi-supervised learning settings**; • **Mathematics of computing** → *Information theory*; • **Information systems** → **Clustering**;

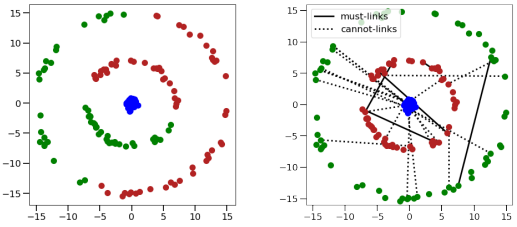
KEYWORDS

semi-supervised clustering, Markov aggregation

ACM Reference Format:
Sophie Steger, Bernhard C. Geiger, and Marek Śmieja. 2022. Semi-Supervised Clustering via Information-Theoretic Markov Chain Aggregation. In *Proceedings of ACM SAC Conference (SAC'22).* ACM, New York, NY, USA, Article 4, 4 pages. https://doi.org/xx.xxx/xxx_x

1 INTRODUCTION

A popular approach to clustering, especially if only pairwise similarities between data points are available, is to view the problem from the perspective of random walks. From this perspective, each data point is represented by a state in the state space of a Markov chain, whose transition probabilities are determined by the pairwise similarities between the corresponding data points. The clustering problem can then be solved via aggregating the state space of the thus defined Markov chain.



(a) (b)

Figure 1: Result for (unsupervised) Markov aggregation clustering (a) and the proposed semi-supervised CoMaC with 30 constraints (b). While also the unsupervised approach can learn non-linear boundaries between clusters, the pairwise constraints help avoiding bad local optima.

Although clustering via Markov aggregation has a solid theoretical basis, allows for creating non-linear decision boundaries, and was shown to achieve competitive performance [1], it appears to be highly sensitive on a careful selection of hyperparameters or optimization procedures. In random walk-based clustering, even representing a dataset as a Markov chain requires selecting hyperparameters, cf. (2) below. Tuning these hyperparameters to individual datasets is cumbersome and severely limits the practical applicability of the respective clustering method. Moreover, there are no clear rules and objective evaluation measures for their selection because of the unsupervised nature of clustering.

In this paper, we propose Constrained Markov Clustering (CoMaC), the extension of clustering via Markov aggregation [2] to the semi-supervised setting, where the side information is given in the form of partition-level information [4–7, 10] (some data points are labeled by their cluster index) or pairwise constraints [8, 9, 13, 14] (for some pairs of data points we know whether they belong to the same or to different clusters, see Figure 1). Experimental results confirm that the proposed adaptions to the Hartigan-style clustering algorithm of [2] achieve performance on common benchmarks that is competitive with the state-of-the-art in semi-supervised clustering. Furthermore, there are indications that introducing side information makes the algorithm more robust to the selection of some of its hyperparameters. Reducing the sensitivity to hyperparameters is important for (semi-supervised) clustering because the limited available information about class labels typically precludes performing proper validation.

SAC'22, April 25 –April 29, 2022, Brno, Czech Republic

ACM ISBN 978-1-4503-8713-2/22/04…$15.00
https://doi.org/xx.xxx/xxx_x

The value of CoMaC is additionally increased by the fact that the considered clustering framework via Markov aggregation unifies previous methods of [1, 12].

Despite these positive results, our experiments also indicate that working with pairwise constraints bears several challenges. We thus discuss potential avenues for future work in the extended manuscript [11], with a focus on the effect of noisy side information and algorithmic aspects of considering cannot- and must-link constraints.

2 SEMI-SUPERVISED CLUSTERING VIA MARKOV AGGREGATION

We aim to cluster elements of a dataset $\mathcal{X} = (x_1, \dots, x_N)$, $x_i \in \mathbb{R}^n$ into K groups or clusters, using a clustering function $g\colon \mathcal{X} \to \{1, \dots, K\}$. Data points within each cluster should have a higher similarity with each other than with those of different clusters, where similarity has to be defined appropriately. Clustering is successful if the candidate clustering function g is close (in a well-defined sense) to the function $g^\bullet\colon \mathcal{X} \to \{1, \dots, K^\bullet\}$ determining the true partition.

Semi-supervised clustering simplifies the task by providing additional information in one of two flavors: First, partition-level side information refers to a subset $\mathcal{X}'$ of $\mathcal{X}$ for which the true cluster indices are known, i.e., $\{(x, g^\bullet(x)) \mid x \in \mathcal{X}' \subset \mathcal{X}\}$. Second, pairwise constraints indicate which pairs of data points of $\mathcal{X}$ must or must not be put in the same cluster; this setting is often referred to as constrained clustering. Pairwise constraints are given as

$$\mathcal{M} = \{ (x, x') \mid g^\bullet(x) = g^\bullet(x') \} \tag{1a}$$

$$\mathcal{N} = \{ (x, x') \mid g^\bullet(x) \neq g^\bullet(x') \} \tag{1b}$$

for a (small) subset of pairs $(\mathcal{M} \cup \mathcal{N}) \subset \mathcal{X}^2$.

Clustering via Markov Aggregation. Identifying each element of the dataset $\mathcal{X}$ with a state of a Markov chain and by parameterizing the transition probability between states via the similarity of corresponding data points, the clustering problem can be formulated as a Markov aggregation problem. For example, if $d\colon \mathcal{X}^2 \to [0, \infty)$ is a measure of dissimilarity between data points, then $\mathcal{X}$ can be clustered via aggregating the Markov chain $X = (X_1, X_2, \dots)$ with state space $\mathcal{X}$ and transition probability matrix $\mathbb{P} = [P_{i,j}]$,

$$P_{i,j} \propto \mathrm{e}^{-\frac{d(x_i, x_j)}{\sigma^2}} \tag{2}$$

where σ^2 is a scaling factor. Indeed, letting $Y = (g(X_1), g(X_2), \dots)$ denote the aggregated process defined via the candidate clustering g, the authors of [1] proposed maximizing the mutual information $I(Y_1; Y_2)$, where $d(x_i, x_j)$ is 0 if x_i and x_j are k-nearest neighbors of each other and ∞ otherwise. In [12], d was chosen as the Euclidean distance, σ^2 as the k-nearest neighbor distance, and the authors proposed to minimize $I(Y_1; X_1) - \beta I(Y_1; X_{T+1})$, where T is selected such that the Markov chain X has relaxed to a meta-stable state.

In this work, we utilize a cost function that has recently been proposed as a generalized information-theoretic framework for Markov aggregation [2]:

$$C_\beta(X, h) = (1 - 2\beta)\,(H(Y_2|Y_1) - H(Y_2|X_1)) - \beta I(Y_1; Y_2). \tag{3}$$

We let the transition probability matrix $\mathbb{P}$ depend on the clustering dataset $\mathcal{X}$ via (2), where we choose d to be the squared Euclidean distance and σ_k^2 as the average squared Euclidean distance between the data point and its k nearest neighbors (averaged over all data points). The approaches of [1] and [12] (for $T = 1$ and symmetric dissimilarity measures d) correspond to solving (3) for $\beta = 0.5$ and for $\beta = 1$, respectively. The main differences between [1, 12] and minimizing (3) rely on the definition of $\mathbb{P}$ in (2) and, potentially, the relaxation time T proposed in [12].

Constrained Markov Clustering (CoMaC). The authors of [2] proposed a Hartigan-style algorithm for minimizing (3) over all deterministic clustering functions $g\colon \mathcal{X} \to \{1, \dots, K\}$. Starting from an initial clustering of $\mathcal{X}$ into K clusters, each data point x is mapped to every aggregate state $y \in \{1, \dots, K\}$ and the cost function is evaluated. The data point is then assigned to the aggregate state that minimizes the cost function.

In this work, the algorithm of [2] is extended in order to accept pairwise constraints $\mathcal{M}$ and $\mathcal{N}$ as given in (1). Since partition-level side information can easily be converted to pairwise constraints (but not vice-versa), the resulting algorithm can handle both types of side information. Below we describe the algorithmic aspects of CoMaC. Pseudocodes of our algorithms are deferred to [11].

Initialization. First, the candidate partition function g is initialized such that all pairwise constraints are satisfied. This is done via solving a graph coloring problem, where no adjacent vertices of a graph are allowed to be of the same color. In our procedure, each vertex of this graph either corresponds to an individual data point not involved in any must-link constraint, or to a set of data points that are connected via must-link constraints, while each edge of this graph corresponds to a cannot-link constraint. The initial coloring of the graph is performed by a greedy algorithm, where each vertex is assigned the first color available in sequence. To avoid the algorithm getting stuck in bad local minima, vertices with no cannot-link constraints are assigned a random color.

Iteration. Once the initial partition function is defined, the sequential algorithm minimizes the cost function in (3) iteratively. Cannot-link constraints are incorporated by restricting the possible states of the aggregation function. Data points connected by must-link constraints are assigned to an aggregate state simultaneously. Erroneous or noisy pairwise constraints can lead to the case where a data point cannot be assigned to any aggregate state. Then, the aggregate state with the least occurrences of cannot-link constraints is selected. CoMaC contains the sequential algorithm in [2] as special case for $\mathcal{M} = \mathcal{N} = \emptyset$. Utilizing the properties of (3), each iteration has a computational complexity of $O(KN^2)$ for clustering a dataset with N elements into K clusters [2, Sec. V].

Constraint Propagation. In this work we assume noise-free, non-conflicting constraints and can therefore *propagate* them to artificially increase the sets of pairwise constraints. Assuming a graph with vertices $\mathcal{X}$ and edges defined by $\mathcal{M}$, we first determine all connected components in this graph using a depth-first search algorithm. Each connected components is then completed to a clique, thus extending $\mathcal{M}$. Additionally, if two data points from different cliques are connected by a cannot-link constraint, then all elements of the two respective cliques are connected by cannot-link constraints, and thus are not allowed to be in the same cluster. E.g., if $(x, x') \in \mathcal{M}$ and $(x, x'') \in \mathcal{N}$, then also x' and x'' should not link, despite (x', x'') not being a labeled cannot-link constraint.

3 EXPERIMENTS

We experimentally evaluate the performance of CoMaC. First, we show that the introduction of pairwise constraints makes Markov aggregation-based clustering less sensitive to hyperparameter settings. We then compare its performance with the state-of-the-art semi-supervised clustering techniques following the experimental setup of [10]. We measure the accuracy of the obtained clusterings with the Normalized Mutual Information (NMI), averaged over 10 randomized runs. Additional results are available in [11][1].

Sensitivity Analysis of CoMaC. In this part, we investigate the effect of hyperparameters on the clustering results produced by CoMaC. To be consistent with [10], we generate pairwise constraints from randomly sampled partition-level side information.

Influence of parameter k. One can observe an influence of the hyperparameter k on the clustering accuracy of CoMaC.[2] Ideally, k is chosen such that the transition probability matrix is nearly completely decomposable, which strongly depends on the chosen dataset. In this subsection, we analyse the influence of k on the three circles dataset shown in Figure 1. Data points are placed uniformly distributed at radii {0.5, 7, 15} and corrupted by spherical Gaussian noise with standard deviation of 0.3.

We analyse the performance of CoMaC with $\beta = \{0,\ 0.5,\ 1\}$ for semi-supervised clustering where 0%, 10%, 20%, 30% of data points are labeled while k is varied (see Figure 2, (a–c)). Indeed, we observe that side information makes CoMaC more robust to the selection of this hyperparameter, at least for this dataset: Clustering accuracy degrades for increasing values of k, and the degradation is less severe the more data points are labeled.

The same experiment is repeated for the Iris dataset (see Figure 2, (d–f)). As it can be seen, the NMI as a function of k shows less variations than for the three concentric circles. As expected, the optimal value of k depends on the dataset. However, for $k < 50$, the performance is quite stable for both datasets and all considered levels of side information. Thus, for all subsequent experiments we set $k = 20$ rather than optimize it for each dataset.

Influence of parameter β. We next analyse the influence of the parameter β for a constant setting of $k = 20$. The results of the experiment for the unsupervised case and for a semi-supervised case where 20% of the data points are labeled and used to generate the pairwise constraints are shown in Figure 4. Unsupervised, CoMaC performs particularly badly for small β values as it is prone to getting stuck in bad local minima. This parallels the behavior of the Markov aggregation method proposed in [2]. To compensate this behavior, the authors proposed an annealing scheme, reducing β iteratively. Since these bad minima for small values of β cannot be escaped by introducing additional side-information, we have implemented this annealing scheme also for CoMaC and report its results in [11]. In this work, we stick with the vanilla version of CoMaC and restrict our attention to $\beta \geq 0.5$, where the algorithm achieves stable results.

Evaluation. Next, we compare CoMaC with the state-of-the-art semi-supervised clustering techniques on several UCI datasets [3]. For a fair comparison, we follow the experimental setup in [10] and directly use the clustering results of comparative methods reported there. We assume that the number of clusters is known.

Experimental setup. We consider CoMaC with $k = 20$ and $\beta = 0.5$ throughout all experiments. The latter parameter setting corresponds to the clustering method proposed in [1], albeit for a different transition probability matrix $\mathbb{P}$.

The following baselines are considered (see [10] for a description of hyperparameters selection): k-means [5] and fuzzy c-means (fc-means, [6, 7]) with partition-level side information, Gaussian Mixture Model (GMM) with partition-level side information (mixmod, [4]), GMM with pairwise constraints (c-GMM, [9]), spectral clustering with pairwise constraints (spec, [8]), and model-based clustering based on cross-entropy and information bottleneck using partition-level side information (CEC-IB, [10]), with two values of a hyperparameter, denoted as CEC-IB$_1$ and CEC-IB$_0$. Since some of the comparison methods reported in [10] only accept partition-level side information, those methods are provided with the ground truth clusters $g^\bullet(x)$ for a subset $\mathcal{X}'$ of data points (0%, 10%, 20%, 30%). This partition-level side information is subsequently converted to pairwise constraints and incorporated to the remaining methods. To allow for a fair comparison, the constraint sets $\mathcal{M}$ and $\mathcal{N}$ are exhaustive, i.e., they contain all pairwise constraints that are implied by partition-level side information. Specifically, if $|\mathcal{X}'| = m$, then $|\mathcal{M}| + |\mathcal{N}| = m(m-1)/2$.

Clustering with side information from all classes. First, we consider a typical case, where the partition-level side information covers elements of all classes. Figure 3 shows the accuracy of the clustering results for each algorithm and dataset for different fractions of labeled data points. Overall, we can observe that CoMaC clearly benefits from labeled data, at least for $k = 20$ and $\beta = 0.5$, as NMI increases with increasing amounts of labeled data points.

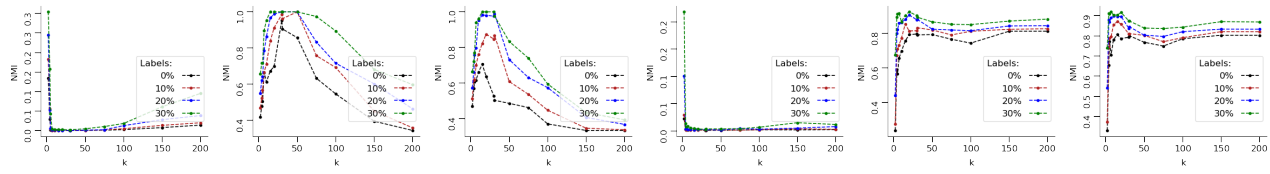
The improvement of CoMaC performance due to side information in comparison to the other techniques is most notable on the Iris dataset. Only 20% of labeled data points noticeably improve the accuracy of CoMaC while the other techniques do not benefit as much from additional side information. CoMaC furthermore achieves superior performance on the Glass, Segmentation, Vertebral and Wine datasets. Both k-means and spec are sensitive to the scale of attributes, which may partially explain why these methods perform worse on the Wine dataset (CoMaC was run on the normalized Wine dataset). Interestingly, on the Vertebral dataset, all algorithms perform equally well in the unsupervised case. However, when incorporating labeled data, both CoMaC and CEC-IB outperform all other methods.

Clustering with side information from a subset of classes. Next, we investigate the case where the side information does not cover all classes. Now, a certain percentage of data points from only two classes is selected and used for labeling. The goal is to determine the ability of our clustering algorithm to correctly identify all classes, although it is given information about only two of them.

The results reported in Table 1 show that CoMaC is robust against missing labels from other classes. The advantage of CoMaC is especially evident in the case of Vertebral, Wine and User dataset, but it also performs well on Glass and Segmentation datasets.

[1] The data and code used for these experiments is publicly available at https://github.com/stegsoph/Constrained-Markov-Clustering

[2] See, e.g., the Section VII.D in the extended preprint of [2] (arXiv:1709.05907).



(a) $\beta = 0$ (b) $\beta = 0.5$ (c) $\beta = 1$ (d) $\beta = 0$ (e) $\beta = 0.5$ (f) $\beta = 1$

Figure 2: Dependence of the clustering accuracy on the parameter k for the circles dataset (a–c) and the Iris dataset (d–f).

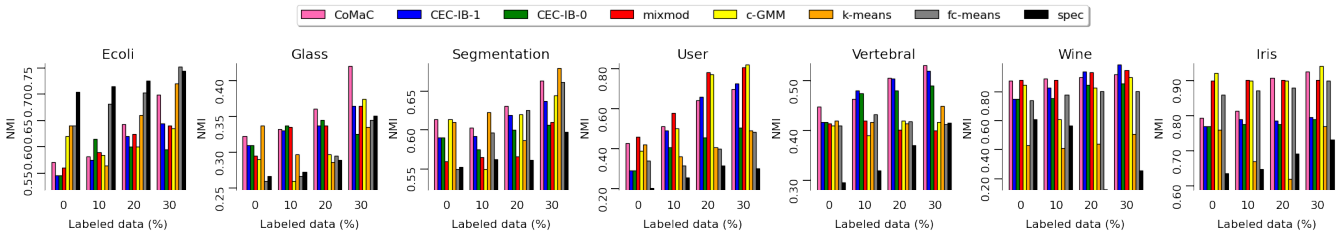


Figure 3: Normalized mutual information computed on UCI datasets with noise-free side information from all classes.

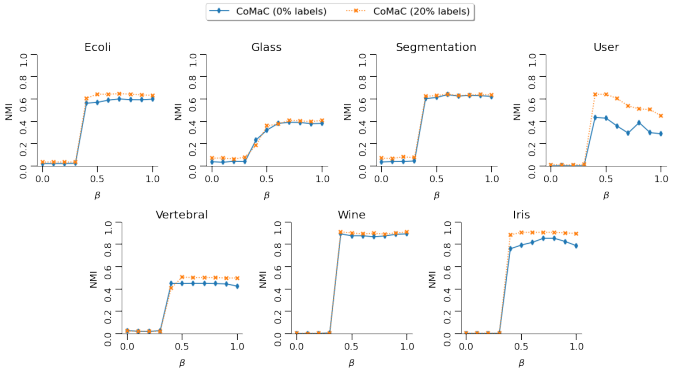


Figure 4: Influence of the parameter β on the clustering accuracy in the unsupervised and semi-supervised setting (fraction of labeled data = 20%).

Table 1: Normalized mutual information (best result is boldface, second in italics) with noise-free information from two classes only (20% of data is labeled, $\beta = 0.5$ and $k = 20$).

	Ecoli	Glass	Segm.	User	Vert.	Wine	Iris
CoMaC	0.6409	**0.4106**	*0.6276*	**0.6482**	**0.5151**	**0.9027**	*0.8484*
CEC-IB-1	0.6050	0.3400	0.6150	*0.4600*	*0.5150*	0.7700	0.7700
CEC-IB-0	0.5900	0.3200	0.5950	0.4440	0.4900	0.7450	0.7850
c-GMM	0.5700	0.3010	0.5500	0.3900	0.3700	0.6600	0.5700
k-means	0.6500	0.2800	**0.6500**	0.3600	0.4170	0.4450	0.6750
fc-means	*0.6892*	0.2703	0.4874	0.4307	0.4319	*0.8063*	**0.8791**
spec	**0.7164**	*0.3624*	0.5714	0.2940	0.4033	0.3690	0.7127

ACKNOWLEDGMENTS

The work of Sophie Steger and Bernhard C. Geiger has been supported by iDev40 (ECSEL Joint Undertaking; grant agreement No. 783163) and the HiDALGO project (H2020 Programme; grant agreement No. 824115). The Know-Center is funded within the Austrian COMET Program. COMET is managed by the Austrian Research Promotion Agency FFG.

REFERENCES

[1] A. Alush, A. Friedman, and J. Goldberger. 2016. Pairwise clustering based on the mutual-information criterion. *Neurocomputing* 182 (2016), 284–293. https://doi.org/10.1016/j.neucom.2015.12.025

[2] R. A. Amjad, C. Blochl, and B. C. Geiger. 2020. A Generalized Framework For Kullback–Leibler Markov Aggregation. *IEEE Trans. Automat. Control* 65, 7 (Jul 2020), 3068–3075. https://doi.org/10.1109/tac.2019.2945891

[3] D. Dua and C. Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[4] R. Lebret, S. Iovleff, F. Langrognet, C. Biernacki, G. Celeux, and G. Govaert. 2014. Rmixmod: The R Package of the Model-Based Unsupervised, Supervised and Semi-Supervised Classification Mixmod Library. *Journal of Statistical Software* 67 (12 2014). https://doi.org/10.18637/jss.v067.i06

[5] H. Liu and Y. Fu. 2015. Clustering with Partition Level Side Information. In *Proc. IEEE Int. Conf. on Data Mining (ICDM)*. IEEE, Atlantic City, NJ, 877–882. https://doi.org/10.1109/ICDM.2015.18

[6] W. Pedrycz, A. Amato, V. Di Lecce, and V. Piuri. 2008. Fuzzy Clustering With Partial Supervision in Organization and Classification of Digital Images. *IEEE Transactions on Fuzzy Systems* 16, 4 (2008), 1008–1026. https://doi.org/10.1109/TFUZZ.2008.917287

[7] W. Pedrycz and J. Waletzky. 1997. Fuzzy clustering with partial supervision. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 27, 5 (1997), 787–795. https://doi.org/10.1109/3477.623232

[8] P. Qian, Y. Jiang, S. Wang, K.-H. Su, J. Wang, L. Hu, and R. F. Muzic. 2017. Affinity and Penalty Jointly Constrained Spectral Clustering With All-Compatibility, Flexibility, and Robustness. *IEEE Transactions on Neural Networks and Learning Systems* 28, 5 (2017), 1123–1138. https://doi.org/10.1109/TNNLS.2015.2511179

[9] N. Shental, A. bar Hillel, T. Hertz, and D. Weinshall. 2003. Computing Gaussian Mixture Models with EM Using Equivalence Constraints.. In *Proc. Advances in Neural Information Processing Systems (NIPS)*. Vancouver.

[10] M. Smieja and B. C. Geiger. 2017. Semi-supervised cross-entropy clustering with information bottleneck constraint. *Information Sciences* 421 (Dec 2017), 254–271. https://doi.org/10.1016/j.ins.2017.07.016

[11] S. Steger, B. C. Geiger, and M. Śmieja. 2021. Semi-Supervised Clustering via Information-Theoretic Markov Chain Aggregation. extended version: `arXiv:2112.09397 [cs.LG]`.

[12] N. Tishby and N. Slonim. 2000. Data Clustering by Markovian Relaxation and the Information Bottleneck Method. In *Proc. Advances in Neural Information Processing Systems (NIPS)*. MIT Press, Cambridge, MA, USA, 619–625.

[13] K. Wagstaff, C. Cardie, S. Rogers, S. Schrödl, et al. 2001. Constrained k-means clustering with background knowledge. In *Proc. Int. Conf. on Machine Learning (ICML)*, Vol. 1. 577–584.

[14] M. Śmieja, O. Myronov, and J. Tabor. 2018. Semi-supervised discriminative clustering with graph regularization. *Knowledge-Based Systems* 151 (2018), 24–36. https://doi.org/10.1016/j.knosys.2018.03.019