Short Paper

Insights into a running clockwork: On interactive process-aware clustering

A Vision Paper

Daniyal Kazempour
Ludwig-Maximilians-Universität München
Munich, Germany
kazempour@dbs.ifi.lmu.de

Thomas Seidl
Ludwig-Maximilians-Universität München
Munich, Germany
seidl@dbs.ifi.lmu.de

ABSTRACT

In recent years the demand for having algorithms which provide not only their results, but also add explainability up to a certain extent increased. In this paper we envision a class of clustering algorithms where the users can interact not only with the input or output but also intercept within the very clustering process itself, which we coin with the term process-aware clustering. Further we aspire to sketch the challenges emerging with such type of algorithms, such as the need of adequate measures which evaluate the progression through the computation process of a clustering method. Beyond the explainability on how the results are generated, we propose methods tailored at systematically analyzing the hyperparameter space of an algorithm, determining in a more ordered fashion suitable hyperparameters rather then applying a trial-and-error schema.

1 INTRODUCTION

Performing a query to the computer science bibliography search engine dblp with the keyword "explainable"[1] delivers an interesting insight looking at the "refine by year" area of the search result as it can be seen in Figure 1 . As for now (November 2018) the number of publications dealing with the aspect of explainability increased from 33 in 2017 up to 101 in 2018. However the scientific works are tailored towards deep learning systems. Since it can be agreed on that deep learning systems have some kind of black box character as stated in e.g. [8], classical clustering methods are fairly transparent regarding the way they generate the clustering results. In the majority of publications dealing with clustering, the whole system can be represented as in Figure 2. Data is given to a clustering algorithm additionally with hyperparameters. The algorithm of choice is executed on that data and yields a clustering result. If clustering methods are already transparent and so easy to comprehend, why should we bother then with the aspect of explainability? Despite the fact that we know how a clustering algorithm works, questions arise like e.g. "what is a good hyperparameter setting?", or "how do my clusters change (or not change) with different hyperparameter settings?", "why did the clusters change that particular way choosing different settings?". Only because we know how the clustering algorithms work, we do not yet fully utilize the potential of this knowledge. Having algorithms like e.g. MeanShift [3] domain experts may want to know what happens with the emerging clusters during the clustering process to understand the resulting clusters. In this vision paper we elaborate in the upcoming sections more in detail on different aspects which are ultimately connected to the idea of including insights from the clustering process itself. For this we address in our vision targets such as points of interaction, hyperparameter analysis, methods and measures as well as potential impacts of our vision on areas such as explainability and didactics. During our tour through these targets we will mention the current related work, and highlight potential difficulties and needs motivating the need for process-aware clustering.

[1] https://dblp.uni-trier.de/search?q=explainable

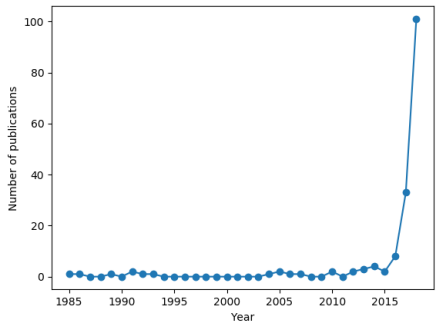

Figure 1: Number of publications indexed at dblp for the keyword explainable from 1985 to 2018.

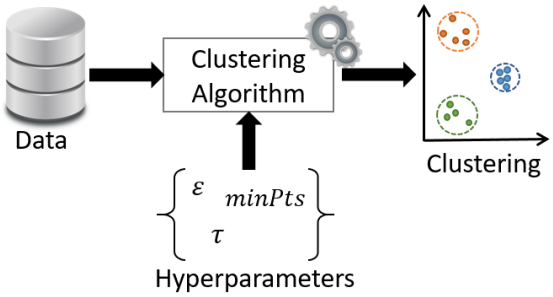

Figure 2: A classic clustering pipeline.

2 INTERACTION TARGETS

In context of interactiveness a rich body of literature is present. Exemplarily we mention iPCA as an interactive tool for PCA-based visual analytics [5]. Despite its high level of sophistication regarding the used visualization techniques, it enables the user to interact with the system *after* a PCA has been performed. The users can not browse through some imtermediate computation steps of PCA and intercept. Also advanced interactive clustering tools like VISA [2] enable human interaction for inspecting the detected clusters and subspaces but do not facilitate to intercept within the clustering process itself. Even in a more recent work

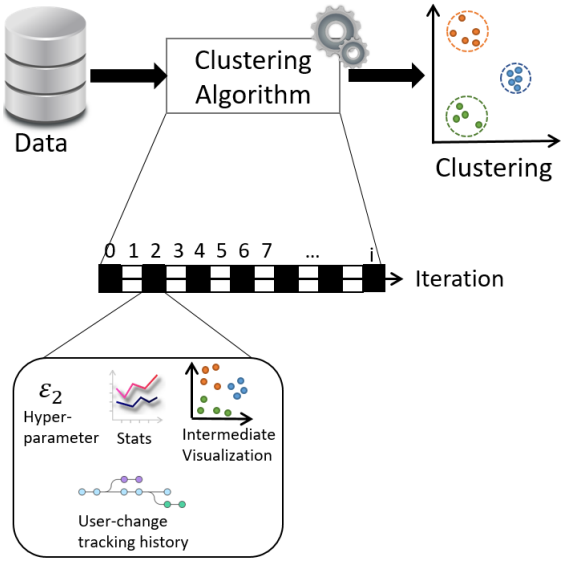

Figure 3: Clustering pipeline in an interactive setting where the users can intercept within each of the steps of the clustering.

[9] , the users can decide on modifying the results in context of hierarchical clustering, yet they do not offer a history of steps which the utilized hierarchical clustering has performed so far.

In a more recent work, a simple but also limited tool PARADISO [6] provides the users the opportunity to explore and intercept within the clustering process itself. In PARADISO our classic clustering pipeline from Figure 2 is re-defined to a pipeline as seen in Figure 3. Here the users can intercept at each of the iterations of a MeanShift algorithm. In this algorithm data points roam within a specific bandwidth (also known as Parzen window) to their mean with regards to the position of other data points located within the bandwidth. The data points thus roam themselves towards their respective mode. In PARADISO the bandwidth hyperparameter can be modified, stats regarding the current clusters are provided, and visualizations for intermediate results at each iteration step are given.

2.1 Multi-instance hyperparameter settings

One aspect from PARADISO which we'd like to emphasize on, is the capability to assign at any iteration step of the algorithm a different bandwidth value, which we coin with the term multi-instance hyperparameter setting, where multi-instance refers to either the iteration step or in general the steps of a clustering algorithm. Going beyond the classical setting where the users provide in the beginning one fixed (set of) hyperparameter(s) which remain(s) valid until the end of an algorithms run, leads us to the case of intercepting within the clustering process and deliberately changing the hyperparameter values at different times. This method becomes even more significant in streaming context. By the simple fact that in a stream setting the data changes over time, a hyperparameter which has been selected in the beginning may no longer be suitable. Thus we may need different hyperparameter values at different times while a clustering algorithm is computing at each iteration on different snapshots of the data.

Further in context of multi-instance hyperparameter settings it is vital to keep track of which changes the users have made. Similar to a version control system, the users shall be given a tool at hand to keep track of the history of changes they have made. By knowing which changes have been made at which iteration, the users can move arbitrarily through the iteration timeline and create alternative branches of changes. Despite giving new targets of interaction, like changing at arbitrary iterations the hyperparameters, the concepts of interactivenes as described so far do bare their own problems which need to be addressed: While on small amounts of data the intermediate computations can be stored, it is in-feasible to store each of the iteration steps at larger data sets. Here potential research targets are e.g. compression strategies and significance measures to determine which of the iteration steps are relevant (enough) to be kept in memory. Such measures could indicate e.g. at which iterations the most changes are taking place.

One question which may arise in context of multi-instance hyperparameter settings is: what are the implications of such an approach? What would it change if we choose different bandwidths *within* a MeanShift run vs. re-running MeanShift multiple times with different bandwidths? Suppose we are given a data set and apply the MeanShift algorithm with a specific bandwidth (e.g. 0.7) in the beginning. Depending on the data set it may happen that a subset of points collapses after a few iterations into one mode. Intercepting a few iterations beforehand by choosing a much smaller bandwidth (e.g. 0.2) may prevent the collapse to one singular mode, leading to multiple modes. Depending at which iteration the interception has taken place, a re-run of MeanShift with a bandwidth of 0.2 may not lead to the same result as in the variant where we start with 0.7 and change it after i-iterations to 0.2. However a proof for this claim is required, which is subject of future works further investigating the multi-instance hyperparameter settings aspect. Yet, we'd like to provide a brief intuition for why it can lead to different results using different bandwidths at different iterations. We envision that MeanShift executed in two instances with each having a bandwidth b_0 and b_1 is like two objects moving among distinct trajectories where time is represented through the single iteration steps. Intercepting in the MeanShift with b_1 is comparable to deflecting the object b_1 leading to a potentially different trajectory than its original path by forces from other modes acting on the object. It may lead to the same destination as MeanShift executed with b_0 but can as well have its destination at a different positions in data space.

3 MEASURING METHODS

Besides the concept of intercepting into the clustering process itself and assigning multiple instances of hyperparameters at different iteration steps, there is a need for measuring methods which are suitable in context of process-aware clustering. The requirements to such measures would be e.g. to capture the dynamics within the clustering process like data points being assigned to specific clusters or subspaces and the change of such over the course of a clustering run.

3.1 Entanglement

The first propositions for such a measure were made in [7]. Here the concept of entanglement has been raised which is intended to support interactive data clustering with the purpose to supply additional information to users. In [7], the entanglement between

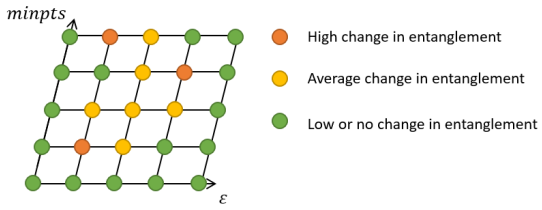

Figure 4: Parameter grid, with the parameters of DBSCAN. Each point represents the entanglement difference computed to its previous entanglement values.

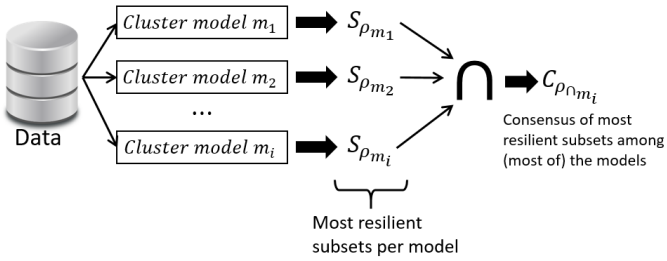

Figure 5: Resilience in context of ensemble clustering.

two data points is defined as the dynamic time warp distance between the trajectories of both points roaming over time to their modes in a MeanShift clustering algorithm. If those data point trajectory pairs are sufficiently similar to each other over the iterations, they are considered as entangled. This definition is however limited to centroid-based approaches like in MeanShift. In our vision we aspire to provide such entanglement definitions for various clustering models (e.g. density-based, hierarchical, spectral, subspace, correlation etc.) and, if possible, a more generalized definition of it.

3.2 Resilience

The entanglement of two data points can be highly varying or remaining stable. This stability depends on the chosen hyperparameters. In [7] the so called resilience has been defined, which computes the variation of the entanglement over different hyperparameter settings. The lower the variance of the entanglement at different hyperparameters, the more resilient is the specific subset of data points. However the work in [7] is highly limited to the MeanShift setting. In our vision auspicious directions include the research of resilience in context of other clustering models, especially also in context of subspaces where the resilience and entanglement is not only determined among subset of data points but also among a subset of subspaces. It remains for future work to investigate in how far the concepts of entanglement and resilience can be applied to different clustering models (e.g. density based, hierarchical, subspace, correlation, spectral etc.) and with which kind of adaptations.

3.3 Hyperparameter Analysis

The research on entanglement and resilience bares potential for the aspect of hyperparameter analysis. In many of the publications on clustering so far, the hyperparameters in e.g. the conducted experiments were chosen on an experience basis. Different settings are tried out until a good (enough) clustering result is achieved. We envision that measures like e.g. resilience can be used together with more systematic approaches in determining the quality of parameters. For example in context of density based clustering like in DBSCAN [4], where we have a minpnts and ϵ parameter, a grid-based approach can prove effective as seen in Figure 4.

Here the granularity of the grid is set by the users on how fine they want to sample the grid. It is advised to start with a coarse grid first. In an initial step, one computes for each of the (minpnts, ϵ) combinations the entanglement. Then the resilience over different parameter settings is computed. Those points on the grid (which represent (minpnts, ϵ) hyperparameters) which have the highest difference in resilience can be considered as interesting, since this specific hyperparameter setting impacts the entanglement of subsets of points. Then in a follow-up step, the grid is refined around such interesting points on the grid. The users can thus successively explore the impact of the hyperparameters. In this context the users can use clusering with a different goal in mind: given a subset of points S_{in} where we want them to be in the same cluster, and further we are given a subset of points S_{not-in} which we do not want to be in the same cluster. Under which parameter settings do we get a clustering fulfilling these constraints? In this setting it is also of interest for future research to approach the scientific problem: how can the users be informed on how "unintuitive" it would be for the clustering algorithm with its underlying model to respect the given constraints as in S_{in} and S_{not-in}?

3.4 Ensemble Settings

The concept of resilience can further be used as a measure for ensembles of clustering models. Given a dataset, in an ensemble setting, the most resilient subsets are considered per model. The intersection of the resulting resilient subsets over all the models yields a subset which is highly resilient among all (or most) of the clustering models which can be seen in Figure 5. The intuition of this intersection subset is, that it is a consensus of different cluster models at different hyperparameter settings, yielding clusters or at least subsets of clusters being valid among different cluster models. On the contrary, removing the consensus subset, leaves subsets which may be (a) not resilient among their respective models, or (b) are highly resilient, but only within their models. The latter case is exciting since this potentially facilitates the extraction of highly *model specific subsets* of data points.

4 IMPACTS

Having elaborated on interaction targets, and measuring methods, we now discuss potential impacts of our vision within the upcoming subsections providing some outlook on the potential magnitudes that this vision may trigger.

4.1 Explainability

Since we have mentioned in the beginning of this vision paper the aspect of explainability we now elaborate on this aspect as a potential impact. The propositions for methods and measures so far enable to look at different parts of an clustering algorithm. The entanglement supports to look into the clustering process itself detecting data points which have the same trajectories or are assigned to the same clusters, subspaces etc. Resilience enables to understand which data points remain (based on entanglement) together even over different hyperparameter settings. The interaction concept of multi-instance hyperparameter setting permits the users to intercept and explore the effects of choosing different values for hyperparameters during the clustering

process. Each of the mentioned methods and aspects can reveal information which can, best to our knowledge, not be gained by simply adding data and some chosen hyperparameter values for a clustering algorithm that once executed only returns the clustering results. However, words of caution are also need to be stated here, since questions that remain open are e.g.: How are the information from the clustering process itself best being presented to the users, and in which form? Which other forms of interaction-tracking may be required, since with increasing number of interaction targets, also the complexity of what can be changed and observed increases.

4.2 Didactics

A connection which may not be obvious on first sight, but becomes rather evident when thinking more from an educational perspective is the relation between explainability and didactics. Besides the theory and exercises in tutorials, demonstrations of the discussed clustering algorithms significantly contribute to the understanding. For such purposes tools like e.g. ELKI [1] exist which can also be used to demonstrate how datasets are clustered and to explore the effects of different hyperparameters by re-running the algorithm every time with a different parameter setting. We are convinced that in our vision process-aware clustering can enable even more insights into the clustering process itself, understanding the behavior, the strengths and also limitations of the process-aware clustering models. It may further aid graduate students which are writing their bachelor or master thesis to evaluate the effects of potential enhancements that they develop and apply to the clustering models they are working with, providing a different approach to evaluate.

5 CONCLUSIONS

In this vision paper we have elaborated on the idea of process-aware clustering, and on its concepts which can be seen summarized in Figure 6. Regarding interaction targets, we have the pillar of multi-instance hyperparameter settings and the pillar of hyperparameter-change tracking with history. While the first pillar enables the interception into the clustering process, the latter provides the capability to track changes and explore different settings. The pillars of entanglement and resilience from the aspect of measuring methods provide the very basis for (a) hyperparameter analysis, which itself serves as the foundation for (b) ensemble settings. All the mentioned fields pose the very foundation for explainability and didactics in the field of interactive process-aware clustering. Further ideas regarding the vision would be to connect process-aware clustering with the research field of process mining. Since various methods are developed for the analysis of processes, some of them (with or without adaptations) may be beneficial to the process-aware clustering concept. Since this vision aims to reveal what happens within the clustering process itself, we conclude this vision paper with a quote from Dr. Faust from a tragic play by Johan Wolfang von Goethe:

That I may understand whatever
Binds the world's innermost core together,
See all its workings, and its seeds,
Deal no more in words' empty reeds.
––Faust, lines 382–385.

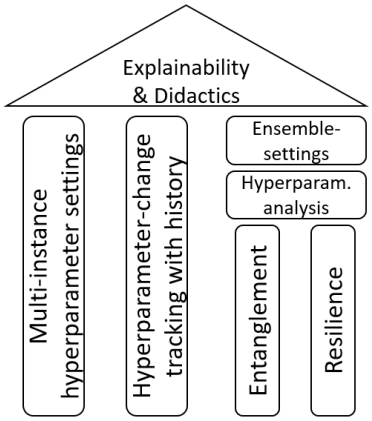

Figure 6: Pillars of process-aware clustering.

ACKNOWLEDGMENTS

This work has been funded by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibilities for its content.

REFERENCES

[1] Elke Achtert, Hans-Peter Kriegel, and Arthur Zimek. 2008. ELKI: A Software System for Evaluation of Subspace Clustering Algorithms. In *Proceedings of the 20th International Conference on Scientific and Statistical Database Management (SSDBM '08)*. 580–585.

[2] Ira Assent, Ralph Krieger, Emmanuel Müller, and Thomas Seidl. 2007. VISA: Visual Subspace Clustering Analysis. *SIGKDD Explor. Newsl.* 9, 2 (Dec. 2007), 5–12.

[3] Yizong Cheng. 1995. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17, 8 (1995), 790–799.

[4] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A Density-based Algorithm for Discovering Clusters a Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*. 226–231.

[5] Dong Hyun Jeong, Caroline Ziemkiewicz, Brian Fisher, William Ribarsky, and Remco Chang. 2009. iPCA: An Interactive System for PCA-based Visual Analytics. *Computer Graphics Forum* (2009).

[6] Daniyal Kazempour, Anna Beer, Johannes-Y. Lohrer, Daniel Kaltenthaler, and Thomas Seidl. 2018. PARADISO: an interactive approach of parameter selection for the mean shift algorithm. In *Proceedings of the 30th International Conference on Scientific and Statistical Database Management, SSDBM 2018, Bozen-Bolzano, Italy, July 09-11, 2018*. 26:1–26:4.

[7] Daniyal Kazempour and Thomas Seidl. 2018. Identifying Entangled Data Points on Iteration Trajectories of Clusterings. In *Proceedings of the Conference "Lernen, Wissen, Daten, Analysen", LWDA 2018, Mannheim, Germany, August 22-24, 2018.* 174–178.

[8] Ravid Shwartz-Ziv and Naftali Tishby. 2017. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810* (2017).

[9] Sharad Vikram and Sanjoy Dasgupta. 2016. Interactive Bayesian Hierarchical Clustering. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*. 2081–2090.