Triclustering Toolbox

Dmitrii Egurnov[1]
(0000-0002-8195-1670) egurnovd@yandex.ru and
Dmitry I. Ignatov[1]
(0000-0002-6584-8534) dignatov@hse.ru

National Research University Higher School of Economics, Moscow, Russia

Abstract. Triclustring Toolbox is a collection of triclustering methods consolidated into a single interface. It provides access to both box- and prime-based OAC (Object-Attribute-Condition) triclustering, Spectral triclustering and features implementations of DataPeeler and Trias. The application also contains algorithms for mining triclusters of similar values: NOAC and Tri-K-Means. Quality of triclusters is measured in terms of density, diversity, coverage, and variance, if applicable. Formats for input and output data of all the methods are universal, which makes comparison and interpretation of the results easier. The code is written in C# (.Net 4.5) and runs on Windows. Triclustring Toolbox was used to provide experimental results in several articles on triclustering.

Keywords: Triadic Formal Concept Analysis · Triclustering · OAC-triclustering · Real-valued context · Software

1 Introduction

Triadic Formal Concept Analysis (3-FCA) was introduced by Lehman and Wille [7] and is aimed at analysis of object-attribute-condition relational data. However, in some cases its strict requirements may be relaxed, that is we could search for less dense structures called triclusters instead of triconcepts [6, 4]. Such patterns found to be useful in several domains, among those mining communities in folksonomies and (multimodal) social networks [6, 5] and semantic frame induction in computational linguistics [9].

The remainder of the paper consists of three sections: Section 2 briefly introduces basic notions, Section 3 describes our Triclustering Toolbox, and Section 4 concludes the paper with future prospects of triclustering software development.

2 Triadic Contexts and Triclusters

Suppose we have a Triadic Formal Context. It has 3 dimensions, or modalities: objects, attributes and conditions.

Definition 1. *Let G, M and B be arbitrary sets. Subset of their Cartesian product defines a triadic relation $I \subseteq G \times M \times B$. The quadruple $\mathbb{K} = (G, M, B, I)$ is called a triadic formal context, or tricontext. The sets G, M and B are called set of objects, set of attributes, and set of conditions, respectively.*

For each triple $(g, m, b) \subseteq I$, where $g \in G$, $m \in M$, and $b \in B$, it is said that "object g has attribute m under condition b". In case of numeric context we add a *value function*: $V : I \to \mathbb{R}$.

Now we define a tricluster in its most general form:

Definition 2. *Let $A_{n \times k \times l}$ be a three-dimensional binary matrix. Let sets $G = \{g_1, g_2, \ldots, g_n\}$, $M = \{m_1, m_2, \ldots, m_k\}$, and $B = \{b_1, b_2, \ldots, b_l\}$ be index sets of A. Then for some arbitrary sets $X \subseteq G$, $Y \subseteq M$ and $Z \subseteq B$ submatrix $A_{XYZ} = \{a_{xyz} \mid x \in X, y \in Y, z \in Z\}$ is called a tricluster. Sets X, Y and Z are called respectively extent, intent and modus of the tricluster.*

Various constraints may be applied to triclusters. Usually they feature structure requirements, cardinality restrictions on extent, intent and modus, and limitations of other parameters. For example, the most common conditions, which eliminate small and meaningless structures from the output, are minimal support condition ($|X| \geq s_X, |Y| \geq s_Y, |Z| \geq s_Z$) and minimum density threshold:

$$\rho(A_{XYZ}) = \frac{\sum_{i=1}^{|X|} \sum_{j=1}^{|Y|} \sum_{k=1}^{|Z|} a_{x_i y_j z_k}}{|X||Y||Z|} \geq \rho_{min}.$$

In case of numeric triclusters we can also measure variance of the values of triples in the tricluster. Lower variance means that values are more similar. If we consider each tricluster $T = (X, Y, Z)$ as an independent sample $S(T) = \{V(g, m, b) \mid (g, m, b) \in I \cap X \times Y \times Z\}$, then unbiased estimate of the variance is:

$$\sigma^2(S) = \frac{\sum_{s \in S} \left(s^2 - \bar{S}^2\right)}{|S| - 1}$$

3 Triclustering Toolbox description

Triclustring Toolbox is a Windows Forms UI application. It was programmed in C# using .Net Framework 4.5. The program takes an input file that contains a triadic context as set of triples. Each triple is described in a separate line with its tab-separated components, which are members of the context's respective modalities. For numerical tricontexts lines also contain the triple's value. The path to the file should be specified in the "Context file" field of the interface. Other parameters are:

- Output folder, where the results and log files of the processing will be placed.
- Limited context uploading. If only several first triples of the context should be processed, the number of triples should be specified.
- Selection of triclustering method:

  - OAC-triclustering [3]
  - Spectral triclustering [3]
  - Box triclustering [8]
  - DataPeeler [1]
  - Trias [6]
  - NOAC [2]
  - Tri-K-Means [2]
- Algorithm-specific options and additional constraints on the output.

When all of the necessary options are set, user can press the "Start" button.

For example, let us look at specific options for numerical triclustering algorithms. For NOAC it is the parameter δ, which is set to 0 by default. It also supports minimal density thresholds for extent, intent, and modus. Tri-K-Means requires the number of clusters k and parameter γ, which defines degree of extending the tricluster over closeness of its values. User can also set upper bound for the number of steps the algorithm takes before terminating and the manner of initialization for centroids, which can be random or predefined. The options are delimited by commas, decimals are separated with a dot. In case several experiments are planned, the sets of options are separated by semicolon.

The program writes two files in the output folder. The first one contains a list of the extracted triclusters along with calculated measures. The first string of the file contains a header, explaining the order of values: *Density, Variance, Average Coverage, Objects coverage, Attributes coverage, Conditions coverage, Extent, Intent,* and *Modus*. Then, in separate lines, the triclusters follow in the format described in the header. Name of the file is composed automatically, using names of the method, input file and the set of options. The second file is a log file. If it already exists in the folder, it will be appended by new execution information. Each line of the file corresponds to a separate experiment and contains the set of options, execution time, number of found triclusters, total coverage of the context by the tricluster set, as well as coverage of extent, intent and modus.

Figure 1 shows an example of Triclustering Toolbox application interface with all described controls.

4 Future work

With recent development of the C# programming language the application may be easily transferred to a cross-platform framework, namely .Net Core 2.0, but only as a console application. UI options are announced in upcoming .Net Core 3.0 release. We would also like to support more recent triclustering methods and provide extension possibilities for other developers.

Another possible important direction is using distributed and parallel computing [10].

Acknowledgements We would like to thank Dmitry V. Gnatyshak for his valuable help with the initial development of the toolbox and Engelbert Mephu

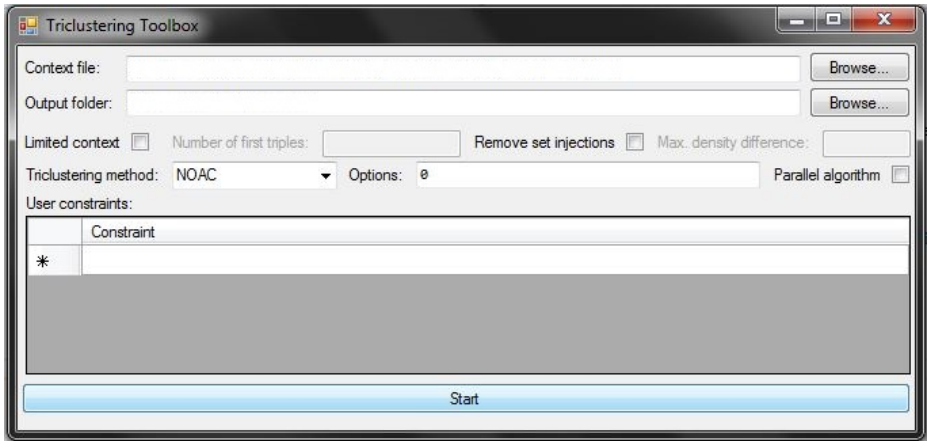

Fig. 1. Triclustering Toolbox User Interface

Nguifo for the inspiration. The work of Dmitry I. Ignatov (contributed to all the sections) was supported by the Russian Science Foundation under grant 17-11-01294 and performed at National Research University Higher School of Economics, Russia.

References

1. Cerf, L., Besson, J., Robardet, C., Boulicaut, J.F.: Data peeler: Contraint-based closed pattern mining in n-ary relations. In: SDM (2008)
2. Egurnov, D., Ignatov, D.I., Nguifo, E.M.: Mining triclusters of similar values in triadic real-valued contexts. In: 14th International Conference on Formal Concept Analysis - Supplementary Proceedings. pp. 31–47 (2017)
3. Ignatov, D.I., Gnatyshak, D.V., Kuznetsov, S.O., Mirkin, B.G.: Triadic formal concept analysis and triclustering: searching for optimal patterns. Machine Learning **101**(1-3), 271–302 (2015). https://doi.org/10.1007/s10994-015-5487-y, http://dx.doi.org/10.1007/s10994-015-5487-y
4. Ignatov, D.I., Kuznetsov, S.O., Magizov, R.A., Zhukov, L.E.: From triconcepts to triclusters. In: Rough Sets, Fuzzy Sets, Data Mining and Granular Computing - 13th International Conference, RSFDGrC 2011, Moscow, Russia, June 25-27, 2011. Proceedings. pp. 257–264 (2011). https://doi.org/10.1007/978-3-642-21881-1_41, http://dx.doi.org/10.1007/978-3-642-21881-1_41
5. Ignatov, D.I., Semenov, A., Komissarova, D., Gnatyshak, D.V.: Multimodal clustering for community detection. In: Missaoui, R., Kuznetsov, S.O., Obiedkov, S.A. (eds.) Formal Concept Analysis of Social Networks, pp. 59–96. Lecture Notes in Social Networks, Springer (2017). https://doi.org/10.1007/978-3-319-64167-6_4, https://doi.org/10.1007/978-3-319-64167-6_4
6. Jäschke, R., Hotho, A., Schmitz, C., Ganter, B., Stumme, G.: TRIAS - an algorithm for mining iceberg tri-lattices. In: Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006), 18-22 December 2006,

Hong Kong, China. pp. 907–911 (2006). https://doi.org/10.1109/ICDM.2006.162, http://dx.doi.org/10.1109/ICDM.2006.162
7. Lehmann, F., Wille, R.: A triadic approach to formal concept analysis. In: Conceptual Structures: Applications, Implementation and Theory, Third International Conference on Conceptual Structures, ICCS '95, Santa Cruz, California, USA, August 14-18, 1995, Proceedings. pp. 32–43 (1995). https://doi.org/10.1007/3-540-60161-9_27, http://dx.doi.org/10.1007/3-540-60161-9_27
8. Mirkin, B.G., Kramarenko, A.V.: Approximate bicluster and tricluster boxes in the analysis of binary data. In: Kuznetsov, S.O., Slezak, D., Hepting, D.H., Mirkin, B.G. (eds.) Rough Sets, Fuzzy Sets, Data Mining and Granular Computing - 13th International Conference, RSFDGrC 2011, Moscow, Russia, June 25-27, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6743, pp. 248–256. Springer (2011). https://doi.org/10.1007/978-3-642-21881-1_40, https://doi.org/10.1007/978-3-642-21881-1_40
9. Ustalov, D., Panchenko, A., Kutuzov, A., Biemann, C., Ponzetto, S.P.: Unsupervised semantic frame induction using triclustering. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers. pp. 55–62. Association for Computational Linguistics (2018), https://aclanthology.info/papers/P18-2010/p18-2010
10. Zudin, S., Gnatyshak, D.V., Ignatov, D.I.: Putting oac-triclustering on mapreduce. In: Yahia, S.B., Konecny, J. (eds.) Proceedings of the Twelfth International Conference on Concept Lattices and Their Applications, Clermont-Ferrand, France, October 13-16, 2015. CEUR Workshop Proceedings, vol. 1466, pp. 47–58. CEUR-WS.org (2015), http://ceur-ws.org/Vol-1466/paper04.pdf