Mining the Groceries Database using Triadic Concept Analysis

Pedro H. B. Ruas[1], Rokia Missaoui[2], Mark A. J. Song[1], Léonard Kwuida[3]

[1] Pontifical Catholic University of Minas Gerais Belo Horizonte, Brazil
pedrohbruas@gmail.com, song@pucminas.br
[2] Université du Québec en Outaouais
rokia.missaoui@uqo.ca
[3] Bern University of Applied Sciences, Bern, Switzerland
leonard.kwuida@bfh.ch

Abstract. In this paper we illustrate the potential of Triadic Concept Analysis (TCA) in extracting useful patterns from triadic formal contexts. To that end, we exploit our recent contributions to TCA to analyze the *Groceries* database and identify triadic patterns expressed by concepts and association rules, including implications.

1 Introduction

Since datasets are frequently expressed by ternary and more generally n-ary relations, one can observe a recent and increasing interest to propose new solutions for the analysis and exploration of these multidimensional data, and specially triadic contexts [2, 4, 6, 7, 10]. This is the case of multidimensional social networks, social resource sharing systems such as folksonomies, and security policies. For instance, in the latter application, a 4-ary or quaternary relation indicates that a *user* is authorized to use *resources* with given *privileges* under restricted *conditions*. For example, User 1000 is allowed to access to Files F_1, F_2 and F_3 with the privilege to *read* F_1 and *update* the last two files in the first three working days of the week.

In this paper, we aim at illustrating the utilization of our software platform using a subset of the well-known dataset named the *Groceries* database [1] of transactions made by customers at a given date for a set of products. Subsets and variants of this dataset have been extensively used by data mining researchers under the name of "market basket analysis" to discover associations between products bought by customers by looking for combinations of items that occur together frequently in customer transactions.

The rest of the paper is organized as follows. In Section 2 we briefly recall the main notions of Triadic Concept Analysis. Section 3 presents our platform, the original dataset and its preprocessing as well as a few examples of the generated patterns which are triadic concepts and association rules, including implications. Finally, Section 4 summarizes the paper and presents the future work in terms of the platform enrichment and validation using synthetic and real data.

RealDataFCA'2021: Analyzing Real Data with Formal Concept Analysis, June 29, 2021, Strasbourg, France

2 Triadic Concept Analysis

Triadic concept analysis was originally introduced by Lehmann and Wille [9] as an extension to FCA, to analyze data described by three sets K_1 (objects), K_2 (attributes) and K_3 (conditions) together with a 3-ary relation $Y \subseteq K_1 \times K_2 \times K_3$. $\mathbb{K} := (K_1, K_2, K_3, Y)$ is called a *triadic context.* A triple (a_1, a_2, a_3) in Y means that object a_1 possesses attribute a_2 under condition a_3. Table 1 shows a row of the context, which partially describes the transactions made by Customer 3782 in K_1 who bought items in K_2, including *pip fruit* in *January*, *chicken* and *oil* on *February*, and *curd* on *January* and *September* as listed in Table 1. Here the condition set K_3 represents the twelve months $(J, F, \ldots, D)$ of the year.

$\mathbb{K}$	Chicken				Oil				Pip Fruit				Curd				...				...			
	J	F	...	S	J	F	...	S	J	F	...	S	J	F	...	S	J	F	...	S	J	F	...	S
3782		1				1			1				1			1	...				...			

Table 1. A row of the triadic context $\mathbb{K} := (K_1, K_2, K_3, Y)$

A *triadic concept* or *closed tri-set* of $\mathbb{K}$ is a triple $c = (A_1, A_2, A_3)$ (also denoted by $A_1 \times A_2 \times A_3$) with $A_1 \subseteq K_1$, $A_2 \subseteq K_2$, $A_3 \subseteq K_3$ and $A_1 \times A_2 \times A_3 \subseteq Y$ is maximal with respect to inclusion in Y. A_1 is called the *extent* of c, A_2 its *intent*, and A_3 its *modus.* We name (A_2, A_3) the *feature* of c. For example, $(\{2512, 4320\}, \{\textit{canned beer}\}, \{\textit{January}, \textit{April}\})$ is one of two triadic concepts (see the green box in Figure 2) with the same extent $\{2512, 4320\}$ which indicates that the two identified customers have two common features. The first feature means that both of them bought *canned beer* in *January* and *April* while the second one indicates that they purchased *butter* and *canned beer* in *January.*

Let $\mathcal{L} = (\mathcal{C}, \leq_1)$ be a poset of nodes such that each node represents the set of triadic concepts in $\mathcal{C}$ with the same extent, and the relation $\leq_1$ is induced by the inclusion on extents. We defined in [10] an adapted version of the *iPred* algorithm to link triadic concepts according to the quasi-order on extents. This allows the Hasse diagram construction of triadic concepts.

A pair (B_2, B_3) is called a triadic feature-based generator [10] (or F-generator) associated with (*i.e.*, compatible with) the feature (A_2, A_3) in a concept (A_1, A_2, A_3) if $A_2 \times A_3 \subseteq (B_2, B_3)^{(1)(1)}$, where the $^{(i)}$-derivation [9] is defined by:

$$X_i^{(i)} := \{(a_j, a_k) \in K_j \times K_k \mid (a_i, a_j, a_k) \in Y \ \forall a_i \in X_i\} \quad (1)$$

$$(X_j, X_k)^{(i)} := \{a_i \in K_i \mid (a_i, a_j, a_k) \in Y \text{ for all } (a_j, a_k) \in X_j \times X_k\}. \quad (2)$$

As an illustration, let us consider the portion of the Hasse diagram shown in Figure 2. We can see that each node represents the set of features and F-generators associated with one extent. For example, there are two features and two F-generators attached to the node whose extent is $\{2512, 4320\}$, The dotted red box attached to the extent $\{2512\}$ contains ten F-generators. One of them is $(\{$*coffee*, *pastry*$\}, \{$*September*$\})$ which is simply denoted by (*coffee, pastry - September*) in Figure 2.

Biedermann [5] defined a *triadic implication* of the form $(A \to D)_C$ with the meaning that "*whenever A occurs under all conditions in C, then D also occurs under the same conditions*". Later on, Ganter and Obiedkov [8] extended Biedermann's definition by proposing three types of implications. We recall two of them in the following.

A *conditional attribute* implication takes the form: $A \xrightarrow{C} D$, where A and D are subsets of K_2, and C is a subset of K_3. It means that *A implies D under all conditions in C*. In particular, the implication holds for any subset in C. This implication is then linked to Biedermann's definition of triadic implication as follows [8]: $A \xrightarrow{C} D \Longleftrightarrow (A \to D)_{C_1}$ for all $C_1 \subseteq C$.

In a dual way, an *attributional condition* implication is an exact association rule of the form $A \xrightarrow{C} D$, where A and D are subsets of K_3, and C is a subset of K_2.

Let (B_2, B_3) be a feature-based generator associated with (A_2, A_3) of a triadic concept whose extent is A_1, i.e., $B_2 \subseteq A_2$ and $B_3 \subseteq A_3$. Then, we define in [10] the *Biedermann conditional attribute implication* (BCAI) as $(B_2 \to A_2 \setminus B_2)_{B_3}$ with a support equal to $|A_1|/|K_1|$ if $B_2 \subset A_2$ and $B_3 \subseteq A_3$, where K_1 stands for the set of objects.

Dually, the *Biedermann attributional condition implication* (BACI) $(B_3 \to A_3 \setminus B_3)_{B_2}$ holds with a support equal to $|A_1|/|K_1|$ if $B_3 \subset A_3$ and $B_2 \subseteq A_2$. A_2 involved in a BCAI is constrained to be maximal among all the intents of the features associated with (B_2, B_3) and A_3 inside a BACI must be maximal among all the modi of the features associated with (B_2, B_3).

Given a feature-based generator $g = (B_2, B_3)$ and a feature (A_2, A_3), both associated with the node whose extent is A_1 such that A_2 is the maximal intent in $((B_2, B_3)^{(1)})^{(1)}$ that contains B_2. Let the quasi-order $(X_1, X_2, X_3) \lesssim_1 (A_1, A_2, A_3)$ holds between the current concept $c = (A_1, A_2, A_3)$ and $c_1 = (X_1, X_2, X_3)$ found in the lower cover of the node that contains c. Then, we claim the following statement:

If $A_2 \subset X_2$ and $X_3 \subseteq A_3$ and $B_2 \subset X_2$ and $B_3 \subseteq X_3$, then the following *Biedermann conditional attribute association rule BCAAR* holds: $(B_2 \to X_2 \setminus A_2)_{B_3}$ with a support equal to $|X_1|/|K_1|$ and a confidence of $|X_1|/|A_1|$, where K_1 stands for the set of objects.

In a dual way, the *Biedermann attributional condition association rule $BACAR$* can be defined.

3 A Case Study

3.1 Platform

Our development platform for TCA is implemented mostly with Python and has the following modules [10] as presented in Figure 1:

1. Call of *Data-Peeler* procedure [6] to get triadic concepts
2. Computation of the precedence links by adapting the *iPred* algorithm [3] to the triadic setting
3. Calculation of two types of triadic generators, including the feature-based generators
4. Computation of association rules, including implications
5. Adaptation of stability and separation indices to the triadic framework.

In this paper, we will mainly illustrate our work on Module 2 and parts of Modules 3 and 4.

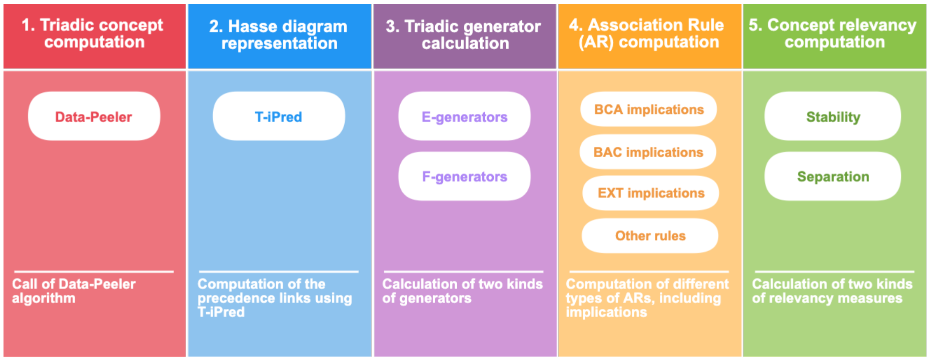
1. Triadic concept computation
Data-Peeler
Call of Data-Peeler algorithm
2. Hasse diagram representation
T-iPred
Computation of the precedence links using T-iPred
3. Triadic generator calculation
E-generators
F-generators
Calculation of two kinds of generators
4. Association Rule (AR) computation
BCA implications
BAC implications
EXT implications
Other rules
Computation of different types of ARs, including implications
5. Concept relevancy computation
Stability
Separation
Calculation of two kinds of relevancy measures

Fig. 1. The architecture modules of our proposed solution.

3.2 Data set

The *Groceries* database [1] contains 38765 transactions, 3898 customers, 167 products (items) and 728 distinct transaction dates. From this database, we extracted and analyzed many samples of different sizes w.r.t. to the number of customers, the number of items and variants of transaction dates (day and month, month only, day of the week, ..). The subset we are analyzing in this paper contains 8121 transactions made by 1000 customers who bought 30 items during a given month (rather than a specific date) between 2014 and 2015. To select items, we took the 23 frequently bought products (after removing the first top 5 ones) and added seven other items which are turkey, chicken, chocolate, meat, ham, ice cream, and napkins. The identified customers are those who bought at least one of the 30 selected products. A portion of the input data after the preprocessing step (but before a conversion into a triadic context) can be seen in Table 2.

The reason we used such a sample is due to the fact that it allows us to illustrate the meaning of the generated patterns that are either triadic concepts, implications or association rules with a confidence lower than 1. Due to the sparsity of the initial dataset and many large subsets, the set of implications was frequently empty or small and many generated association rules were trivial with a very low support.

The sample is then converted into a triadic context where the value 1 in a cell indicates that a given customer bought an item at least once during a given month. For example, Customer 3782 purchased a pip fruit on January as shown in Table 1.

Costumer number	**Item**	**Month**
2512	butter	January
2512	coffee	September
2512	bottled_water	September
2512	domestic_eggs	April
2512	root_vegetables	January
2512	fruit/vegetable_juice	September
2512	newspapers	September
2512	bottled_water	April
2512	ham	April
2512	domestic_eggs	January
2512	canned_beer	January
2512	canned_beer	April
2512	pastry	September

Table 2. Input data.

3.3 Patterns

In the following we show a part of the output produced for the subset of the *Groceries* database by first displaying a part of the Hasse diagram and then a set of implications and association rules.

Figure 2 shows a portion of the Hasse diagram of triadic concepts where the node in the green box represents the extent of two triadic concepts as a set of two customers who share two features found inside the corresponding blue box. The first feature tells us that Customers 2512 and 4320 bought the item *canned beer* on *April* and *January* while the second feature indicates that they both purchased *butter* and *canned beer* on *January*. Since the set of F-generators is equal to the set of features associated with the extent $\{2512, 4320\}$, no implication can be generated from the node labeled with this extent. If we look at the upper covers of this current node, we observe three groups of customers: those who bought *canned beer* in *January* (winter season), those who purchased *canned beer* in *April* (spring) and those who bought *butter* in *January*.

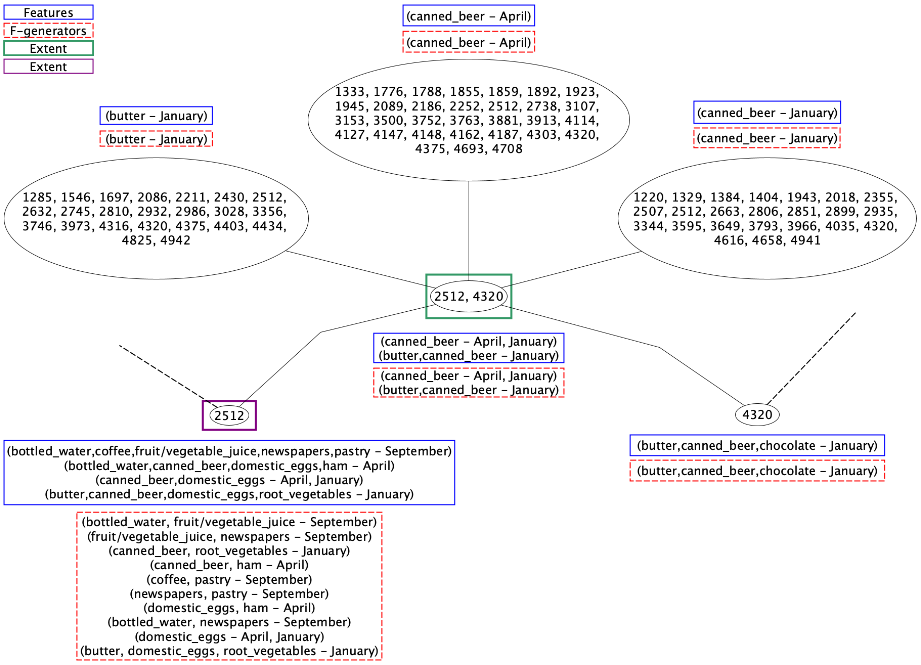
Features
F-generators
Extent
Extent
(canned_beer – April)
(canned_beer – April)
1333, 1776, 1788, 1855, 1859, 1892, 1923, 1945, 2089, 2186, 2252, 2512, 2738, 3107, 3153, 3500, 3752, 3763, 3881, 3913, 4114, 4127, 4147, 4148, 4162, 4187, 4303, 4320, 4375, 4693, 4708
(butter – January)
(butter – January)
1285, 1546, 1697, 2086, 2211, 2430, 2512, 2632, 2745, 2810, 2932, 2986, 3028, 3356, 3746, 3973, 4316, 4320, 4375, 4403, 4434, 4825, 4942
(canned_beer – January)
(canned_beer – January)
1220, 1329, 1384, 1404, 1943, 2018, 2355, 2507, 2512, 2663, 2806, 2851, 2899, 2935, 3344, 3595, 3649, 3793, 3966, 4035, 4320, 4616, 4658, 4941
2512, 4320
(canned_beer – April, January)
(butter,canned_beer – January)
(canned_beer – April, January)
(butter,canned_beer – January)
2512
(bottled_water,coffee,fruit/vegetable_juice,newspapers,pastry – September)
(bottled_water,canned_beer,domestic_eggs,ham – April)
(canned_beer,domestic_eggs – April, January)
(butter,canned_beer,domestic_eggs,root_vegetables – January)
(bottled_water, fruit/vegetable_juice – September)
(fruit/vegetable_juice, newspapers – September)
(canned_beer, root_vegetables – January)
(canned_beer, ham – April)
(coffee, pastry – September)
(newspapers, pastry – September)
(domestic_eggs, ham – April)
(bottled_water, newspapers – September)
(domestic_eggs – April, January)
(butter, domestic_eggs, root_vegetables – January)
4320
(butter,canned_beer,chocolate – January)
(butter,canned_beer,chocolate – January)

Fig. 2. Portion of the Hasse diagram annotated with extents, features and F-generators.

The node in the purple box concerns Customer 2512 with four features displayed in the blue box and 10 F-generators in the red box. We can then extract the following BCAI using the first feature and the first F-generator:
$(bottled\ water, fruit/vegetable\ juice \rightarrow coffee,\ newspapers, pastry)_{September}$ $[0.1\%, 100\%]$. It means that whenever this customer (one out of $1000 = 0.1\%$) buys *fruit/vegetable juice* and *bottled water* at least once on *September*, then he/she purchases *coffee*, *newspapers* and *pastry* during this month. In a similar way, we can identify the following BACI from another node in the diagram (not shown in Figure 2):
$(February, April \rightarrow November)_{root\ vegetables}$ $[2\%, 100\%]$. It means that buying *root vegetables* on *February* and *April* implies making the same purchase on *November* with a support of 2%.

All the empirical tests are executed using multi-threading on a Ubuntu 19.10 based system with 32GB of RAM memory and an Intel i7-4790 3.6GHz 8-core processor.

In Table 3 we present a few statistics about the size of the output as well as the execution time of each one of the platform component.

Statistics	Groceries dataset
Nb. of links	11124
Nb. of distinct extents	3128
Nb. of triadic concepts	3912
Nb.of minimal F-generators	5124
Nb. of implications (sup >0)	2164
Nb. of association rules (conf. <1)	7290
Execution time in seconds per step	
T-iPred	2.524
F-generator computation	**95.721**
Implication computation	0.053
Association rule computation (conf. <1)	16.281
Total time	**114.579**

Table 3. Statistics and execution times in seconds for the sample (1000 × 30 × 12) of the *Groceries* dataset.

4 Conclusion and Further Development

In this paper we illustrated the application of algorithms for Triadic Concept Analysis to a subset of the *Groceries* database as a triadic context to discover concepts and association rules, including implications. The merits of TCA lie in the fact that patterns are in a compact form and under many perspectives in the sense that for a given extent (set of objects) of a triadic concept, we obtain different views expressed by distinct features, and for a given generator and compatible features, we may get more than one association rule or implication.

Our current work is to complete the development of our software solution to make it an open source for everyone and continue our investigation of new kinds of association rules [10] on other datasets.

Finally, in order to help researchers validate their present and future contributions in TCA and more generally in Polyadic Concept Analysis [4, 6, 11], the FCA research community members need to join their forces and share their best experiences in collecting, exchanging and using clean and coherent multidimensional datasets. Software tools to generate synthetic data of different sizes and density levels are also welcome to evaluate the performance and scalability of the designed algorithms.

Acknowledgment

This study was financed in part by the NSERC (Natural Sciences and Engineering Research Council of Canada) and by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Finance Code 001.

References

1. Groceries dataset for market basket analysis. https://www.kaggle.com/heeraldedhia/groceries-dataset
2. Ananias, K.H., Missaoui, R., B. Ruas, P.H., Zarate, L.E., Song, M.A.: Triadic concept approximation. Information Sciences (2021)
3. Baixeries, J., Szathmary, L., Valtchev, P., Godin, R.: Yet a Faster Algorithm for Building the Hasse Diagram of a Concept Lattice. In: ICFCA'09. pp. 162–177 (2009)
4. Bazin, A.: On implication bases in n-lattices. Discrete Applied Mathematics 273, 21–29 (2020), advances in Formal Concept Analysis: Traces of CLA 2016
5. Biedermann, K.: How triadic diagrams represent conceptual structures. In: ICCS. pp. 304–317 (1997)
6. Cerf, L., Besson, J., Robardet, C., Boulicaut, J.: Data peeler: Contraint-based closed pattern mining in n-ary relations. In: Proceedings of the SIAM International Conference on Data Mining, SDM 2008, April 24-26, 2008, Atlanta, Georgia, USA. pp. 37–48. SIAM (2008)
7. Felde, M., Stumme, G.: Triadic exploration and exploration with multiple experts. CoRR abs/2102.02654 (2021)
8. Ganter, B., Obiedkov, S.: Implications in triadic formal contexts. In: Wolff, K.E., Pfeiffer, H.D., Delugach, H.S. (eds.) Conceptual Structures at Work. pp. 186–195. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)
9. Lehmann, F., Wille, R.: A triadic approach to formal concept analysis. In: ICCS. pp. 32–43 (1995)
10. Missaoui, R., Ruas, P.H., Kwuida, L., Song, M.A.: Pattern discovery in triadic contexts. In: ICCS. pp. 117–131. Springer (2020)
11. Voutsadakis, G.: Polyadic concept analysis. Order 19(3), 295–304 (2002)