

Análise Formal de Conceitos Triádicos através da utilização de Diagramas Binários de Decisão

Kaio H. A. Ananias¹, Julio C. V. Neves¹, Luis E. Zárate¹, Mark A. J. Song¹

Pontifícia Universidade Católica de Minas Gerais (PUC-MG), Brazil
deandradekaio@gmail.com, juliocesar.neves@gmail.com, zarate@pucminas.br, song@pucminas.br

Abstract. Formal Concept Analysis (FCA) is an approach based on the mathematization and hierarchy of formal concepts. Nowadays, with the increasing of social network for personal and professional usage, more and more applications of data analysis on environments with high dimensionality (Big Data) have been discussed in the literature. Through the Formal Concept Analysis and Triadic Concept Analysis, it is possible to extract database knowledge in a hierarchical and systematized representation. It is common that the data set transforms the extraction of this knowledge into a problem of high computational cost. Therefore, this work has an objective to evaluate the behavior of the algorithm for extraction in order to extract triadic concepts using TRIAS with high dimensional contexts. It was used a synthetic generator known as SCGaz (Synthetic Context Generator a-z). After the analysis, it was proposed a representation of triadic contexts using a structure known as Binary Decision Diagram (BDD).

Categories and Subject Descriptors: H.2.8 [Database Applications]: Data Mining

Keywords: Formal Concept Analysis, Triadic Concept Analysis, Data Mining, Binary Decision Diagram

1. INTRODUÇÃO

A descoberta de informação válida, tácita, compreensível e útil é o objetivo de diversas áreas de conhecimento na Ciência da Computação. A dificuldade em atingir essa meta se agrava conforme essas bases se tornam cada vez maiores. Dentre os desafios está o problema de encontrar relações e regras que descrevam o comportamento dos elementos presentes. Tome, como exemplo, a crescente popularização das redes sociais e o volume de dados produzidos por seus usuários. Esta é uma aplicação que cria uma demanda por técnicas para a extração de conhecimento de forma a explicitar as interações entre usuários e definir padrões que representem o comportamento da rede.

Uma possível solução para o problema é a utilização da Análise Formal de Conceitos (AFC) que é uma técnica baseada na matematização da noção de conceitos e na estruturação destes em uma hierarquia conceitual. Com o uso da AFC é possível a análise dos dados através de associações e dependências de objetos e atributos formalmente descritos a partir de um conjunto de dados real ou sintético [Wille 1982] [Ganter and Wille 1999]. A representação do conhecimento contido na base se faz via descrição dos objetos, atributos e das relações de incidência entre os mesmos denominada contexto formal. Nesta abordagem tradicional, denominada diádica, representa-se a informação por meio de uma tripla (G, M, I) , em que G é o conjunto de objetos, M o conjunto de atributos e I a relação binária de incidência entre G e M .

Em diversas situações, entretanto, é necessário descrever a condição que estabelece a relação entre os diferentes objetos e seus atributos. Uma extensão da AFC clássica (diádica), denominada Análise Formal de Conceitos Triádicos (TCA) foi proposta com o objetivo de lidar com este problema [Lehmann and Wille 1995]. Apesar de oriunda da AFC a abordagem triádica é mais complexa por lidar

Copyright©2018 Permission to copy without fee all or part of the material printed in KDMiLe is granted provided that the copies are not made or distributed for commercial advantage, and that notice is given that copying is by permission of the Sociedade Brasileira de Computação.

com dados tridimensionais. A TCA é baseada na relação triádica entre objetos, atributos e condições definida pela quádrupla (K_1, K_2, K_3, Y) em que K_1, K_2 e K_3 são, respectivamente, os conjuntos de objetos, atributos e condições e Y a relação ternária entre os mesmos.

Assim como na AFC, a abordagem triádica tem que lidar com problemas em que as bases de dados são de alta dimensionalidade. Embora diversos algoritmos tenham sido propostos na literatura com intuito de extrair informação de conceitos triádicos, nenhum ataca diretamente o problema de alta dimensionalidade [Jaschke et al. 2006] [Cerf and Besson 2009] [Trabelsi et al. 2012].

Desta forma, o objetivo deste trabalho consistiu em: 1) avaliar o comportamento de algoritmo em bases triádicas de alta dimensionalidade (especificamente o TRIAS [Jaschke et al. 2006]); 2) propor modificações no gerador sintético de contextos diádicos SCGaz para que o mesmo seja utilizado na geração de contextos triádicos [Rimsa et al. 2013]; 3) gerar contextos sintéticos que possibilitassem a análise do comportamento do algoritmo TRIAS para extração de conceitos triádicos (buscou-se compreender o comportamento deste algoritmo quando submetido ao processamento de bases de dados de alta dimensionalidade) e, 4) representar contextos triádicos utilizando *BDDs* (*Binary Decision Diagram*) para armazenar e manipular de forma eficiente contextos de alta dimensionalidade [Akers 1978]. Neste caso implementou-se um conjunto de operações *booleanas* para a recuperação de objetos, atributos e condições.

O artigo está assim dividido: a seção seguinte apresenta a fundamentação teórica, a seção 3 os trabalhos relacionados, a seção 4 a abordagem, os testes e análises, e por fim a seção 5, as conclusões e trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Desenvolvida por Rudolf Wille na década de 80, a Análise Formal de Conceitos (AFC) é um ramo da matemática aplicada baseada na matematização do conceito e da hierarquia conceitual [Wille 1982] [Ganter and Wille 1999]. A AFC considera os conceitos como meios de compreensão intersubjetiva em situações de ação orientada para o propósito. A formalização dos conceitos deve ser transparente e simples, mas também abrangente, de modo que os aspectos principais de um conceito possam ter suas referências explícitas no modelo formal [Lehmann and Wille 1995].

A abordagem diádica se baseia na noção primitiva de um contexto formal que é uma tripla (G, M, I) , em que G é o conjunto de objetos, M o conjunto de atributos e I a relação binária de incidência entre G e M , indicando que um objeto g de G possui um certo atributo m de M . A Tabela I representa um contexto diádico. Conceitos formais e regras de implicação podem ser extraídos de contextos diádicos.

Table I. Contexto Diádico representado por uma tabela cruzada

| G/M | m_1 | m_2 | m_3 |
|-------|-------|-------|-------|
| g_1 | × | × | |
| g_2 | | × | × |
| g_3 | × | | |

Um conceito formal de um contexto formal (G, M, I) é definido por um par (A, B) em que $A \subseteq G$, $B \subseteq M$. O par (A, B) que define o conceito segue as condições $A = B'$ e $B = A'$ definida pelo operador de derivação ($'$): $A' = \{g \in G \mid gIm \forall m \in B\}$ e $B' = \{m \in M \mid gIm \forall g \in A\}$ - a extensão A contém cada objeto de G que possui todos os atributos de B , e a extensão B contém todos atributos de M pertencentes a todos objetos de A .

2.1 Análise Formal de Conceitos Triádicos (TCA)

A TCA introduzida por Lehmann and Wille [Lehmann and Wille 1995], estende a AFC clássica com a inserção de uma nova dimensão. A noção primitiva de um contexto formal triádico é definida por uma quádrupla (K_1, K_2, K_3, Y) em que K_1, K_2 e K_3 são conjuntos e Y a relação ternária entre K_1, K_2 e K_3 . Os elementos de K_1, K_2 e K_3 são chamados objetos, atributos e condição respectivamente e

$(o_1, a_2, c_3) \in Y$ é interpretado como o objeto o_1 que possui o atributo a_2 sob a condição c_3 [Lehmann and Wille 1995] [Wille 1995]. A Tabela II apresenta um contexto triádico onde as incidências são representadas através da relação entre os objetos o_i , atributos a_i e conceitos c_i do contexto, marcadas ou não, com por uma cruz.

Table II. Contexto Triádico representado por uma tabela cruzada

| K_1/K_2-K_3 | c_1 | | | c_2 | | | c_3 | | |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | a_1 | a_2 | a_3 | a_1 | a_2 | a_3 | a_1 | a_2 | a_3 |
| o_1 | × | × | | | × | | × | | |
| o_2 | | | × | | | × | | × | × |
| o_3 | × | | | × | | | | × | |

Um conceito formal triádico é definido por uma tripla (A_1, A_2, A_3) , tal que $A_1 \subseteq K_1$, $A_2 \subseteq K_2$ e $A_3 \subseteq K_3$ e $A_1 \times A_2 \times A_3 \subseteq Y$. Os conjuntos A_1 , A_2 , A_3 são chamados objetos, atributos e modo, respectivamente [Lehmann and Wille 1995]. O conjunto de todos os conceitos de um contexto triádico parcialmente ordenado formam um reticulado completo denominado reticulado conceitual [Missaoui and Kwuida 2011].

2.2 Gerador Sintético SCGaz

Utilizar base de dados sintéticas para geração de contextos formais torna-se interessante devido a complexidade das bases de dados obtidas a partir de cenários reais. Bases de dados reais geralmente necessitam de um pré-processamento, tarefa esta que pode, se não for efetuada corretamente, interferir diretamente nos resultados.

A ferramenta *SCGaz* proposta em [Rimsa et al. 2013] é um gerador sintético randômico de contextos formais diádicos irredutíveis com controle de densidade. Através do *SCGaz* é possível especificar a quantidade de objetos e atributos desejados em um contexto formal, bem como a densidade, para a geração de contextos.

2.3 Algoritmo TRIAS

Em [Jaschke et al. 2006] os autores definem o problema de minerar todos os conceitos triádicos mais frequentes de um contexto formal e propõe uma solução chamada TRIAS baseada em projeções diádicas para resolução. Os autores adaptam a noção diádica de minerar todos os *itemsets* de um contexto formal diádico, definida em [Pei et al. 2000], para o caso triádico. Seja $\mathcal{K} = (U, T, R, Y)$ um contexto formal triádico e $u\text{-mimsup}$, $t\text{-mimsup}$ e $r\text{-mimsup} \in [0, 1]$. O problema de extrair todos os conceitos triádicos frequentes de um contexto consiste em determinar todas as triplas (A, B, C) do contexto \mathcal{K} tal que $\frac{|A|}{|U|} \geq u\text{-mimsup}$, $\frac{|B|}{|T|} \geq t\text{-mimsup}$ e $\frac{|C|}{|R|} \geq r\text{-mimsup}$.

Em [Jaschke et al. 2006] os autores introduzem o algoritmo TRIAS para computar todos os conceitos triádicos frequentes de um contexto formal/*folksonomia*. Seja $\mathcal{K} = (U, T, R, Y)$ um contexto triádico, o algoritmo TRIAS primeiro constrói um contexto diádico $\mathcal{L} = (U, T \times R, Y_1)$ onde suas colunas correspondem a pares de elementos pertencentes a T e R e via projeção, extrai todos os conceitos formais. O segundo passo consiste em, para cada conceito formal, checar se estes são fechados em relação a U . A característica principal do algoritmo está em explorar os subconjuntos de conceitos triádicos recém computados para verificar se estes são novos conceitos.

2.4 Diagrama Binário de Decisão

Introduzido por [Akers 1978] e desenvolvido posteriormente por [Bryant 1986], os diagramas binários de decisão (*BDD*) fornecem uma representação canônica para fórmulas *booleanas* muito mais compactas que formas normais conjuntivas e disjuntivas além de proporcionarem eficiência em sua manipulação. É possível obter um *BDD* a partir de uma árvore de decisão binária, em que os traços pontilhados representam transições nulas, ou seja com valor 0 e os traços contínuos representam transições positivas

com valor 1 - exemplo na Figura 1. A ideia principal dos diagramas binários de decisão está em mesclar subárvores da árvore binária de decisão e eliminar nós idênticos (redundantes) resultando em representação canônica. O resultado das otimizações nos fornece um grafo acíclico direcionado.

3. TRABALHOS RELACIONADOS

Diversos trabalhos usam *BDDs* com diferentes objetivos. Em [Salleb et al. 2002] os *BDDs* foram usados para armazenar logs de transação como uma tabela de verdade e encontrar padrões frequentes em grandes conjuntos de dados transacionais. O uso de *BDDs* permitiu que os autores carregassem todas as transações em memória principal, evitando processamento da base de dados em disco.

Em [Neto et al. 2018] os autores utilizam o diagrama binário de decisão para lidar com contextos diádicos de alta dimensionalidade na extração de conceitos formais diádicos. Os autores propõem modificações nos algoritmos *NextClosure* e *In-Close2* através de *BDD* para a manipulação dos objetos de um contexto diádico. Para os testes foram utilizados o algoritmo *NextClosure* e contextos diádicos com até 50.000 objetos e 25 atributos, gerados por meio da ferramenta *SCGaz*. Os autores obtiveram ganhos significativos com o uso de *BDD*, obtendo resultados de até 4 vezes melhor que a implementação original. Além disso, o uso dos diagramas permitiu que os autores explorassem contextos com maior dimensionalidade, como 50.000 objetos, 20 atributos com 70% de densidade. Os autores também exploraram o algoritmo *In-Close2* com uso de *BDD*. Mais uma vez a abordagem se mostrou eficiente em diversos casos. Os autores obtiveram *speedup* de até 2 em contextos com 500.000 objetos e máxima densidade. Em diversas situações a versão com *BDD* foi capaz de gerar conceitos enquanto o algoritmo original encerrava a execução devido a estouro de memória.

Em [Santos et al. 2018] os autores propõem modificações no algoritmo de extração de implicações próprias diádicas *ProperIm*, adicionando *BDDs* para manipulação e extração das regras em contextos diádicos. O algoritmo *ProperImplicBDD* apresenta tempos de execução significativamente melhores. Os testes variaram a quantidade de atributos e densidade dos mesmos para um total de 120.000 objetos. Os resultados mostraram que a versão utilizando *BDD* gerou ganhos significativos no tempo de execução, chegando a ser 4 vezes mais rápido que a versão original. Além de ganho de performance, os autores conseguiram explorar contextos, por exemplo, com 150 atributos e densidades de 30%, 50% e 70% (maior dimensionalidade) que a versão original do algoritmo não era capaz de processar, expandindo o horizonte de aplicações.

4. A ABORDAGEM, TESTES E ANÁLISES

Este trabalho teve por objetivo analisar o comportamento do algoritmo TRIAS em contextos triádicos de alta dimensionalidade gerados a partir de uma ferramenta sintética (*SCGaz*) para geração de contextos triádicos. Foi proposta também uma representação de contextos triádicos com uso de *BDDs*. Tal abordagem pode ser utilizada em trabalhos futuros como estrutura principal de algoritmos triádicos como TRIAS, conforme explicitado em seções anteriores.

Modificações no gerador de contextos sintético diádicos *SCGaz* foram realizadas para geração de conceitos triádicos adicionando uma terceira dimensão, não computada pela ferramenta. As regras de redutibilidade definidas em [Rimsa et al. 2013] são mantidas para os contextos triádicos. Uma terceira dimensão escolhida pelo usuário é adicionada e os objetos do contexto diádico são então replicados para o contexto triádico sujeitos às condições previamente definidas.

A partir das modificações anteriores, avaliações sobre o comportamento do algoritmo TRIAS foram realizadas utilizando-se dos contextos anteriormente gerados pelo *SCGaz*. Tempo médio de execução, quantidade de conceitos encontrados, dimensões e densidade, são avaliadas nos testes iniciais deste trabalho com intuito de encontrar os limites do algoritmo. Através de projeções do contexto triádico em diádico, funções booleanas são geradas a partir do contexto e em seguida o *BDD* é construído.

4.1 Contextos Triádicos usando SCGaz

O gerador sintético *SCGaz* proposto em [Rimsa et al. 2013] fornece uma abordagem diádica para geração de contextos randômicos. Contudo, a abordagem triádica TCA, possui uma terceira dimensão comumente chamada de Condições. Esta dimensão fornece uma maior caracterização de objetos, uma vez que estes agora estão relacionados com um determinado atributo sob uma condição.

Neste trabalho, estendeu-se a ferramenta *SCGaz* adicionando a dimensão de condições nos contextos gerados. A quantidade de condições é definida pelo usuário. Dado um contexto formal irreduzível (G, M, I) , gerado por meio do *SCGaz*, uma incidência diádica é definida por $gIm \subseteq I$, em que $g \in G$ e $m \in M$. Um contexto triádico (K_1, K_2, K_3, Y) é gerado em que $K_1 = G$ e cada atributo $a_i \in K_2$ é definido por:

$$a_i = m_i \bmod |K_3|$$

Dada uma incidência $gIm \in I$, em que $g \in G$ e $m \in M$, do contexto formal irreduzível (G, M, I) , obtido por meio do *SCGaz*, um contexto triádico (K_1, K_2, K_3, Y) é gerado, e a regra que adiciona a incidência gIm atrelada à condição c_i é definida por:

$$c_i = \frac{m_i}{|K_3|}$$

4.2 Avaliação do Algoritmo TRIAS

A partir do gerador sintético randômico *SCGaz*, diversos contextos foram gerados a fim de avaliar o comportamento do algoritmo de extração de conceitos triádicos TRIAS. Contextos triádicos sintéticos com número arbitrário de dimensões e densidade foram gerados com o intuito de entender o comportamento do algoritmo.

Inicialmente, fixou-se a quantidade de atributos e condições, maximizando a quantidade de objetos com intuito de obter um maior número de incidências. Contextos com 500, 1.500, 3.000, 5.000 e 10.000 objetos foram gerados com 15 atributos e 5 condições. A densidade foi fixada em 30% para todos os contextos pois, o principal objetivo era compreender as dimensões limites para o algoritmo TRIAS (a quantidade de atributos, objetos e condições suportadas).

Os testes foram executados em um Intel Core i7-4790 3.60GHz com 4 cores, 8 *threads* 32Gb de memória RAM e um sistema operacional Ubuntu 14.04 LTS. A Tabela III apresenta os resultados considerando inicialmente contextos com dimensões reduzidas segundo [Old and Priss 2006]. É possível notar que mesmo com um número reduzido de objetos, atributos e condições, o algoritmo leva aproximadamente 40 minutos para computar todos os conceitos do primeiro contexto sintético. Observe que o teste com 10.000 objetos, 15 atributos e 5 condições demandou um tempo superior a 7 dias e não foi devidamente computado conforme mostrado.

Table III. Resultados do Algoritmo TRIAS para contextos menores

| Contexto (Objetos X Atributos X Condições) | Incidências | TRIAS (Minutos) |
|---|-------------|--------------------|
| 500 x 15 x 5 | 13500 | 42.68 |
| 1500 x 15 x 5 | 33750 | 212.4 |
| 3000 x 15 x 5 | 67500 | 376.2 |
| 5000 x 15 x 5 | 112500 | 768.8 |
| 10000 x 15 x 5 | 225000 | - |

É interessante notar que as características de alta dimensionalidade em aplicações triádicas podem diferir das mesmas diádicas investigadas. Em 2006 a edição do *International Conference on Formal Concept Analysis (ICFCA)* em *Desdren* [Old and Priss 2006] discutiu os principais desafios da análise formal, dentre eles, a necessidade de lidar com contextos formais densos e de alta dimensionalidade, por exemplo, 120.000 objetos e 70.000 atributos, estes consideravelmente maior que os testes aqui observados.

Novos testes foram realizados, com contextos caracterizados de alta dimensionalidade. A Tabela IV mostra os resultados obtidos com contextos de 120.000 objetos, variando atributos e condições respectivamente. É possível notar que em nenhum dos casos de alta dimensionalidade conseguiu-se chegar ao fim da execução do algoritmo, mantendo-o executando por mais de 7 dias sem nenhuma conclusão.

Os resultados obtidos na execução do TRIAS evidenciam o custo computacional elevado da extração de conhecimento de contextos triádicos. É notória a inviabilidade de se utilizar TCA com bases de dados de alta dimensionalidade. Este fato certamente demanda uma maior investigação e novas propostas para viabilizar o uso de algoritmos de TCA neste contexto.

Table IV. Resultados do Algoritmo TRIAS para contextos de alta dimensionalidade

| Contexto | | | Incidências | TRIAS (Dias) |
|--------------|-------------|-----------|-------------|-----------------|
| (Objetos X | Atributos X | Condições | | |
| 120.000 x 15 | x 5 | | 2.999.984 | >7 |
| 120.000 x 10 | x 5 | | 1.776.769 | >7 |
| 120.000 x 5 | x 10 | | 1.776.769 | >7 |

4.3 Contextos Triádicos usando *BDD*

Algumas aplicações de AFC utilizam *BDD* como estruturas principais para armazenamento e manipulação de objetos de forma eficiente [Salleb et al. 2002] [Neto et al. 2018] [Santos et al. 2018]. Desta forma, este trabalho propõe uma representação triádica de contextos usando *BDDs* com intuito de tirar proveito da eficiência e economia proporcionada por estes diagramas.

Dado um contexto formal triádico (K_1, K_2, K_3, Y) em que K_1, K_2 e K_3 são chamados objetos, atributos e condição respectivamente e Y a relação ternária entre K_1, K_2 e K_3 , uma projeção pode ser realizada no contexto triádico (Tabela V) resultando em um contexto diádico $(K_1, K_2 \times K_3, Y)$ (Tabela VI). A projeção resulta da combinação dos atributos e condições, de forma que cada atributo

Table V. Contexto Triádico (K_1, K_2, K_3, Y)

| K_1/K_2-K_3 | c_1 | | c_2 | | c_3 | |
|---------------|-------|-------|-------|-------|-------|-------|
| | a_1 | a_2 | a_1 | a_2 | a_1 | a_2 |
| o_1 | x | | | x | x | |
| o_2 | | x | x | | | x |
| o_3 | x | | x | | x | x |

seja renomeado de acordo com a condição a qual pertence. A recuperação e manipulação dos atributos e condições podem ser feitas a partir do *label* atribuído a cada atributo. No contexto representado pela Tabela VI a incidência diádica dada pela tupla $(o1, a1c1)$ é equivalente a incidência triádica dada pela tripla $(o1, a1, c1)$ do contexto representado na Tabela V. Uma vez projetado, o contexto

Table VI. Projeção Diádica do Contexto $(K_1, K_2 \times K_3, Y)$

| $K_1/K_2 \times K_3$ | a_1c_1 | a_2c_1 | a_1c_2 | a_2c_2 | a_1c_3 | a_2c_3 |
|----------------------|----------|----------|----------|----------|----------|----------|
| o_1 | x | | | x | x | |
| o_2 | | x | x | | | x |
| o_3 | x | | x | | x | x |

triádico agora descrito por um contexto diádico pode ser representado por um diagrama binário de decisão convertendo o contexto para uma fórmula booleana usada para gerar o *BDD* correspondente. A Tabela VI descreve o contexto triádico projetado em um contexto diádico e a através de operações conjuntivas e disjuntivas entre os atributos do contexto o mesmo é representado pela Equação 1, onde os símbolos acentuados por uma barra representam a negação do atributo.

$$f(a_1c_1, a_2c_1, a_1c_2, a_2c_2, a_1c_3, a_2c_3) = a_1c_1 \cdot a_2\bar{c}_1 \cdot a_1\bar{c}_2 \cdot a_2c_2 \cdot a_1c_3 \cdot a_2\bar{c}_3 + a_1c_1 \cdot a_2\bar{c}_1 \cdot a_1\bar{c}_2 \cdot a_2c_2 \cdot a_1c_3 \cdot a_2\bar{c}_3 + a_1c_1 \cdot a_2\bar{c}_1 \cdot a_1\bar{c}_2 \cdot a_2c_2 \cdot a_1c_3 \cdot a_2\bar{c}_3 \quad (1)$$

A Figura 1 representa a projeção diádica do contexto triádico definido anteriormente, onde os nós incididos por arestas pontilhadas significam uma transição nula, ou seja, o objeto não possui o atributo representado pelo nó. Tal abordagem permite manipular contextos triádicos utilizando um *BDD*.

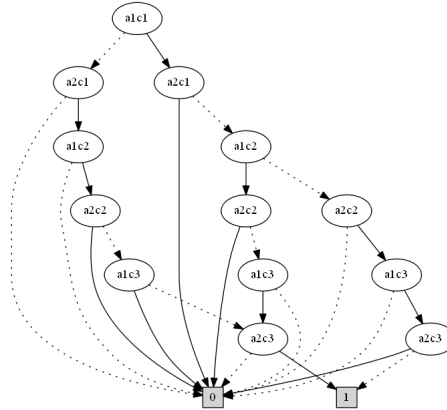


Fig. 1. Contexto $(K_1, K_2 \times K_3, Y)$ representada por um *BDD*.

Dado um contexto triádico projetado e representado por um *BDD*, é interessante prover técnicas para recuperação de objetos, atributos e condições, uma vez que qualquer aplicação que faça uso desta representação necessitará de meios que permitam a recuperação e alteração eficiente destes elementos.

Considere o contexto representando na Tabela V, a recuperação de objetos pode ser feita, por exemplo, a partir de operações lógicas *AND* ou *OR* sob a Equação 1 do contexto. Caso seja necessário obter todos os objetos do contexto representado na Tabela VI que possuam o atributo a_1c_2 , pode-se criar um objeto *BDD* que represente tal atributo e aplicar uma operação lógica *AND* entre os *BDDs*.

Em algumas situações, caso seja necessário recuperar todos os objetos que possuam, por exemplo, os atributos a_1c_1 e a_1c_3 do contexto representado na Tabela VI, um *BDD* com ambos atributos deve ser criado e a operação lógica *AND* entre este novo *BDD* com o *BDD* do contexto deve ser efetuada. Isto retornará apenas os objetos de interesse. A Figura 2 ilustra tal operação.

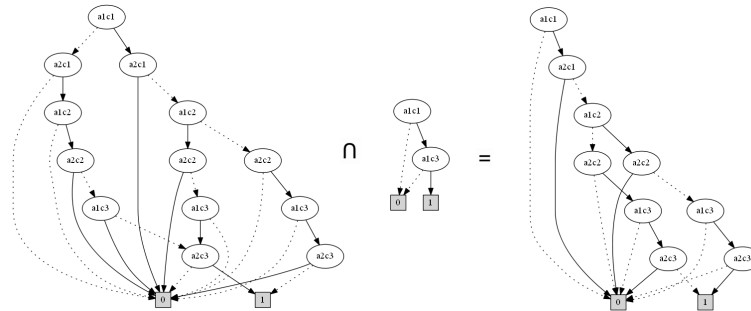


Fig. 2. Operação lógica entre o atributo a_1c_1 e a_1c_3 e o *BDD* Contexto.

5. CONCLUSÕES E TRABALHOS FUTUROS

A tarefa de extrair os conceitos de um contexto triádico mostra-se mais complexa do que na abordagem clássica da FCA. A representação dos dados em três dimensões acarreta na maior dimensionalidade das bases de dados, de forma que com o crescimento dos contextos, técnicas como o algoritmo TRIAS se tornam ineficientes na extração de informação como mostrado. Como intuito de melhorar o desempenho destes algoritmos, a representação de contextos por meio de estruturas como *BDD* se mostrou uma alternativa interessante e eficiente na recuperação de informação.

Através do estudo realizado, percebe-se que o problema de alta dimensionalidade em contextos triádicos acontece com um reduzido número de objetos, atributos (e condições) quando comparado à abordagem diádica. Os testes efetuados mostraram que o algoritmo TRIAS, por exemplo, não consegue lidar com as dimensões caracterizadas como de alta dimensionalidade diádicas e se mostrou ineficiente quando utilizado com contexto de maior dimensão para a abordagem triádica.

Por sua vez, a representação triádica de contextos por meio de *BDDs* aqui proposta mostrou-se eficiente na recuperação das informações relevantes de um contexto tornando-se uma abordagem interessante para aplicações triádicas. A aplicação desta técnica se torna interessante não apenas para prolongar o uso dos algoritmos triádicos em contextos de alta dimensionalidade, mas também para prover eficiência na manipulação de objetos, atributos e condições.

Como trabalho futuro, pretende-se implementar uma versão *BDD* do algoritmo TRIAS. O objetivo é reduzir o tempo das consultas realizadas a fim de classificar o subconjunto de conceitos recém descobertos e conseqüentemente aumentar o poder de extração de conceitos triádicos mais frequentes em um contexto triádico. Espera-se também reduzir o tempo de execução tendo em vista que os resultados apresentados na Tabela IV se mostraram inviáveis.

6. AGRADECIMENTOS

Os autores agradecem o suporte financeiro da CAPES, FAPEMIG e CNPq.

REFERENCES

- Akers, S. B. (1978). Binary decision diagrams. *IEEE Transactions on Computers*, C-27(6):509–516.
- Bryant, R. E. (1986). Graph-based algorithms for boolean function manipulation. *Computers, IEEE Transactions on*, 100(8):677–691.
- Cerf, L. and Besson, J. and Robardet, C. B. J. (2009). Closed patterns meet n-ary relations. pages 1–36.
- Ganter, B. and Wille, R. (1999). Formal concept analysis: mathematical foundations.
- Jaschke, R., Hotho, A., Schmitz, C., Ganter, B., and Stumme, G. (2006). Trias—an algorithm for mining iceberg tri-lattices. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 907–911. IEEE.
- Lehmann, F. and Wille, R. (1995). A triadic approach to formal concept analysis. *Conceptual structures: applications, implementation and theory*, pages 32–43.
- Missaoui, R. and Kwuida, L. (2011). Mining triadic association rules from ternary relations. *Formal Concept Analysis*, pages 204–218.
- Neto, S. M., Zárate, L. E., and Song, M. A. (2018). Handling high dimensionality contexts in formal concept analysis via binary decision diagrams. *Information Sciences*, 429:361–376.
- Old, J. and Priss, U. (2006). Some open problems in formal concept analysis. problems presented at international conference on formal concept analysis (icfca).
- Pei, J., Han, J., Mao, R., et al. (2000). Closet: An efficient algorithm for mining frequent closed itemsets. In *ACM SIGMOD workshop on research issues in data mining and knowledge discovery*, volume 4, pages 21–30.
- Rimsa, A., Song, M. A., and Zárate, L. E. (2013). Scgaz—a synthetic formal context generator with density control for test and evaluation of fca algorithms. In *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, pages 3464–3470. IEEE.
- Salleb, A., Maazouzi, Z., and Vrain, C. (2002). Mining maximal frequent itemsets by a boolean based approach. In *European Conf. on Artificial Intelligence, Lyon France (July 2002)*, pages 285–289.
- Santos, P., Neves, J., Silva, P., Dias, S. M., Zárate, L., and Song, M. (2018). An approach to extract proper implications set from high-dimension formal contexts using binary decision diagram.
- Trabelsi, C., Jelassi, N., and Yahia, S. B. (2012). Scalable mining of frequent tri-concepts from folksonomies. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 231–242. Springer.
- Wille, R. (1982). Restructuring lattice theory: An approach based on hierarchies of concepts. pages 445–470.
- Wille, R. (1995). The basic theorem of triadic concept analysis. *Order*, 12(2):149–158.