Proceedings of the 17th International Conference
on Computational and Mathematical Methods
in Science and Engineering, CMMSE 2017
4–8 July, 2017.

Closed sets enumeration: a logical approach.

F. Benito-Picazo[1], **P. Cordero**[1], **M. Enciso**[1] **and A. Mora**[1]

[1] *Universidad de Málaga, Andalucía Tech, Málaga, Spain*

emails: fbenito@lcc.uma.es, pcordero@uma.es, enciso@lcc.uma.es, amora@ctima.uma.es

Abstract

Closed sets are the basis for the development of the concept lattice, a key issue in formal concept analysis. The enumeration of all the closed sets is a complex problem, having an exponential cost. In addition to the closed set, it is very useful for applications to add the information of all the minimal generators for each closed set. In this work we explain how to approach this problem from a complete set of implication by means of a sound and complete logic.

Key words: Formal concept analysis, closed sets, minimal generator, logic.

1 Introduction

Formal concept analysis (FCA) is a theoretical and practical framework to store information and manage them [GW99]. Data is stored in a table, representing a binary relation between a set of objects and attributes. The success of FCA relies on its solid theoretical framework and a wide set of methods and techniques to extract the knowledge from this data and manipulate it. One outstanding representation of the knowledge is the concept lattice, built over the closed sets, considering the subset relation as the order relation. Such representation depict a overall view of the information with a very strong formalism, opening the door to use the lattice theory as a metatheory to manage the information [BDVG17].

If-then rules have been introduced in several areas, dressed with different clothes. Thus, in relational databases [Cod71] they are named Functional Dependencies, in FCA they are named Implications and in Logic Programming (fuzzy logic) [BV06a] they are named if-then rules. All this notions captures a very intuitive idea: when the premise occurs, then the conclusion holds. Nevertheless, their semantics are very different and they further use are also distinct. In this work we consider implications as elements to describe the information and we design a method to enumerate all closed sets and their minimal generators.

The proposed method is an evolution of [CEMO12], where the authors introduce a logic-based method based on $\mathbf{SL}_{FD}$, a sound and complete logic for implications. That method works by traversing the set of implications and applying a set of inference rules, following a tree paradigm in its execution. In that method, an exhaustive search was developed, producing the intended result but with an improvable performance. Here, we propose the design of several pruning strategies to improve such performance. These strategies are motivated by the idea of avoiding the opening of full branches in the tree or reducing the size of the information in their nodes.

The rest of the work is organized as follows: in the following section we present $\mathbf{SL}_{FD}$ and the axiomatic system which constitutes the basis of the MinGen algorithm. In Section 3 we present the algorithm to enumerate the closed sets and minimal generators and summarize the strategies to improve its execution in practice. The work ends with a brief conclusion.

2 Logic for implications

First, the main notions related formal concept analysis needed in this work are presented.

A **formal context** is a triple $\mathbf{K} := (G, M, I)$ where G is a set of objects, M is a set of attributes and $I \subseteq G \times M$ is a binary relation between G and M such that, for $o \in G$ and $a \in M$, $o\ I\ a$ means that the object o has the attribute a. Then, two mappings are defined $(\ )'\colon 2^G \to 2^M$ defined for all $A \subseteq G$ as $A' = \{m \in M \mid g\ I\ m \text{ for all } g \in A\}$, and $(\ )'\colon 2^M \to 2^G$ defined for all $B \subseteq M$ as $B' = \{g \in G \mid g\ I\ m \text{ for all } m \in B\}$. We use the same symbol since no confusion arises. This pair of mappings is a Galois connection.

The composition of the intent and the extent mappings, and vice versa, introduces two closure operators $(\ )''\colon 2^G \to 2^G$ and $(\ )''\colon 2^M \to 2^M$. The notion of closed set (as a fixpoint of a closure operator) is defined as follows:

Definition 1 *A* ***formal concept*** *is a pair (A, B) such that $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. Consequently, A and B are closed sets of objects and attributes, respectively called extent and intent.*

In this work we focus on the attributes closed sets. A key point in this work is the notion of the minimal generator (mingen) [GW99], which provides a minimal representation for each closed set, and is defined as follows:

Definition 2 *Let $\mathbf{K} = (G, M, I)$ be a formal context and $A \subseteq M$. The set of attributes A is said to be a minimal generator (**mingen**) if, for all set of attributes $X \subseteq A$ if $X'' = A''$ then $X = A$.*

 ISBN: 978-84-617-8694-7

Remark that the above definition allows to characterize each closed set by means of a minimal subset to provide a canonical representation of the closed sets. Moreover, we would like to remark that such representation is not unique, since a given closed sets can have several minimal generators.

The notion of minimal generator can also be defined from the point of view of implications. They are expressions $A \to B$ where A and B are attribute sets. A context satisfies the implication $A \to B$ if every object that has all the attributes from A also has all the attributes from B.

Definition 3 *An (attribute) implication of a formal context* $\mathbf{K} = (G, M, I)$ *is defined as a pair* (A, B)*, written* $A \to B$*, where* $A, B \subseteq M$ *and* $A \cap B = \varnothing$*. Implication* $A \to B$ *holds (is valid) in* $\mathbf{K}$ *if* $A' \subseteq B'$*.*

The set of all valid implications in a context satisfies the well-known Armstrong's axioms [Arm74], which constitutes the pioneer logic to manage implications. The author introduces a sound a complete axiomatic system to infer new implications holding in a context from a given set of implications. Moreover, this logic constitutes a proposal to solve the attribute closure, i.e. to find the maximal set of attributes A^+ such that the implication $A \to A^+$ holds. As we mentioned, this maximal set is a closed set as defined before and this closure operator $()^+$ allows us to guide the automatization the search for closed sets. Thus, we introduce a new logic suitable for this goal.

The introduction of the Simplification Logic [MECF12], named $\mathbf{SL}_{FD}$, opened the door to the development of automated reasoning methods directly based on its novel axiomatic system. $\mathbf{SL}_{FD}$ considers reflexivity as axiom scheme

$$\texttt{[Ref]} \quad \frac{}{A{\to}A}$$

together with the following inference rules called Fragmentation, Composition and Simplification respectively.

$$\texttt{[Frag]} \ \frac{A{\to}BC}{A{\to}B} \qquad \texttt{[Comp]} \ \frac{A{\to}B,\ C{\to}D}{AC{\to}BD} \qquad \texttt{[Simp]} \ \frac{A{\to}B,\ C{\to}D}{A(C \setminus B){\to}D}$$

Similarly to the dual vision of closed set (in terms of Galois connection and implications), a dual definition of minimal generator can be done in terms of implications:

Definition 4 *Let* $\mathbf{K} = (G, M, I)$ *be a formal context and* $A \subseteq M$*. The set of attributes* A *is said to be a minimal generator (**mingen**) if, for all set of attributes* $X \subseteq A$ *if* $X{\to}A^+$ *then* $X = A$*.*

In the following section we introduce the algorithm to enumerate the minimal generator based on $\mathbf{SL}_{FD}$ and describe the strategies to improve its performance.

 ISBN: 978-84-617-8694-7

3 An algorithm to enumerate all closed sets and their minimal generators

Simplification logic has allowed us to design several executable methods to manage implications. Thus in [MECF12] we developed a novel method to compute attribute closure strongly based on $\mathbf{SL}_{FD}$ inference rules. This method has been showed to have a better performance than the classical methods based on indirect techniques. One outstanding characteristics of $\mathbf{SL}_{FD}$ closure is the output it renders: given a set of attributes $X \subseteq M$ and a set of implications Γ, it renders its closure X^+ and a new set of implications Γ' which describes the remaining knowledge in the set $M \setminus X^+$.

This logic-based closure method is the basis of another method, named MinGen, to compute the set of all minimal generators from a set of implicant set presented in [CEMO12]. The algorithm works by applying the $\mathbf{SL}_{FD}$Closure algorithm to each implication in the set, opening a new branch. This application provides a new candidate to be added to mingen and a smaller implications set which guides us in the search of new sets of attributes to be added to mingens, producing a tree-like execution.

In summary, the input of this algorithm is a set of attributes M and a set of implications Γ over the attributes in M. The output is the set of closed sets endowed with all the minimal generators, i.e. $\{\langle C, mg(C)\rangle : C \text{ is a closed set of attributes}\}$ where $mg(C) = \{D : D \text{ is a mingen and } D^+ = C\}$. In this work we only consider non-trivial minimal generators, i.e. pairs of closed set and minimal generator $\langle X, Y\rangle$ where $Y \subsetneq X$.

For example, if $M = \{a, b, c, d, e, f\}$ and $\Gamma = \{a{\to}b, bc{\to}d, de{\to}f, ace{\to}f\}$ the output is the set $\{\langle abcdef, \{ace\}\rangle, \langle abdef, \{ade\}\rangle, \langle abcde, \{ac\}\rangle, \langle bcdef, \{bce\}\rangle, \langle bcd, \{bc\}\rangle, \langle def, \{de\}\rangle, \langle ab, \{a\}\rangle, \langle c, \{c\}\rangle, \langle \varnothing, \{\varnothing\}\rangle\}$. The execution of the method is depicted in Figure 1. We refer the reader to [CEMO12] for a detailed description of the method and its theoretical results.

In this work, we propose two pruning strategies to improve MinGen method. We briefly describe them as follows:

- The first strategy characterizes the branches that can be considered a superfluous one because all their nodes explore closed sets and minimal generators already considered in another branches. To implement this strategy we will consider a subset test on the branches of the same level and in the same branch.

- The second strategy is to expedite the execution of the method by including the closure method in each node in two steps, so that in the first one the closure set is computed and, in the second one, the resulting set of implications is computed taking into account this closed set. In this way, the resulting set of implications will be a smaller one and the method will have a better performance.

 ISBN: 978-84-617-8694-7

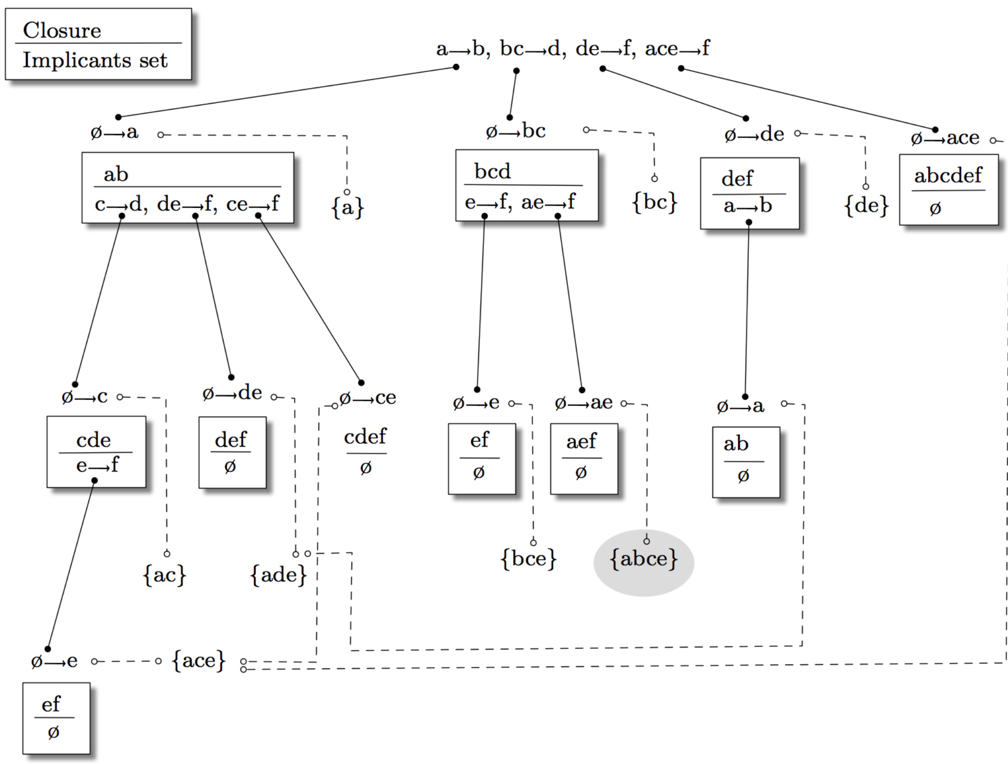

Figure 1: Exemplification of

 ISBN: 978-84-617-8694-7

4 Conclusion and future works

In this work we have studied the state of the art in the enumeration of closed sets and minimal generators based on logic. We have considered the MinGen method based on Simplification Logic as the target of our work and we propose to improve it by means of several prunes to improve its performance.

In a future work, we propose to establish the theoretical results to state these strategies and to develop an exhaustive practical experiment to show its benefits.

Acknowledgements

This work is partially supported by project TIN2014-59471-P of the Science and Innovation Ministry of Spain, co-funded by the European Regional Development Fund (ERDF).

References

[Arm74] W. Armstrong. Dependency structures of data base relationships. *Proceedings of the International Federation for Information Processing Congress*, pages 580–583, 1974.

[BV06a] R. Belohlávek and V. Vychodil. Computing non-redundant bases of if-then rules from data tables with graded attributes. In *2006 IEEE International Conference on Granular Computing*, pages 205–210, 2006.

[BDVG17] K. Bertet, C. Demko, J.-F. Viaud, and C. Guérin. Lattices, closures systems and implication bases: A survey of structural aspects and algorithms. *Theoretical Computer Science*, `http://dx.doi.org/10.1016/j.tcs.2016.11.021`, 2017.

[Cod71] E. F. Codd. Further normalization of the data base relational model. *IBM Research Report, San Jose, California*, RJ909, 1971.

[MECF12] A. Mora, M. Enciso, P. Cordero, and I. Fortes. Closure via functional dependence simplification. *International Journal of Computer Mathematics*, 89(4):510–526, 2012.

[CEMO12] P. Cordero, M. Enciso, A. Mora, and M. Ojeda-Aciego. Computing minimal generators from implications: a logic-guided approach. In *Proceedings of the Ninth International Conference on Concept Lattices and Their Applications*, pages 187–198, 2012.

[GW99] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations.* Springer-Verlag New York, Inc., 1999. Translator-C. Franzke.