A Method based on Association Rules to Construct Product Line Model

Alberto Lora-Michiels, Camille Salinesi, Raul Mazo

A Method Based on Association Rules to Construct Product Line Models

Alberto Lora-Michiels[1, 2] Camille Salinesi[2] and Raúl Mazo[2,3]
[1] *Baxter International Inc, Lessines-Belgium*
[2] *CRI, Panthéon Sorbonne University, 90, rue de Tolbiac, 75013 Paris, France.*
[3] *Ingeniería de Sistemas, Universidad de Antioquia, Medellín-Colombia.*
albertoloram@gmail.com, camille.salinesi@univ-paris1.fr, raulmazo@gmail.com

Abstract—The success of a product line is the ability to improve application engineering, heavily depends on the quality of Product Line Models (PLMs). This paper reports on our effort to develop a method that exploits mining techniques such as the apriori algorithm, independence tests and the like to automate the construction of a PLM specified with FORE, starting from a collection of Product Models (PMs). Using these techniques, the proposed method guides the identification of candidate features, group cardinalities and dependencies. These can be used to progressively construct the PLM consistently with the existing PMs. The method was developed and tested in an industry setting starting with bills of materials as a collection of PMs. One interesting lesson learn from this experiment is that while the PLM is constructed, the domain engineer discovers errors in PMs. We believe that this advocates for a tighter intertwining between domain engineering and application engineering.

I. INTRODUCTION

Product Line (PL) based development is a promising approach to develop software intensive systems in a reuse approach. Promises are multiple: reduced time to market, lower development costs, more trustworthy products, etc.

Approaches to construct PLMs are often focused on using clustering methods to elicitate, prioritize and triage requirements. Rather than a systematic process, the construction of an initial product line model from product requirement specifications somehow remains a "black art" and still mostly relies on the experience and expertise of domain engineers. Proposing new methods, techniques and tools that guide the construction of PLMs is thus a challenge [1].

Our method starts with a collection of PMs and produces PLMs in the FORE [2] notation. The method starts by arranging features of the collection of product models into a matrix of occurrences. Then, the process guides the construction of the general tree architecture by detecting candidate father-son dependencies, mandatory and optional relationships and completes it with group cardinalities. Last it guides the identification of other dependencies such as requires and excludes. The domain of statistics provides several mining techniques that could be used to support this process [3], [4], [5]. The research challenge was thus to identify which techniques could be used to efficiently detect the target items at each step of the method.

Our research strategy was to experiment the available techniques on a real case. Once a technique was detected, further work was needed to identify with which parameter it should be used (e.g. thresholds). Last the overall method was evaluated on the complete case to identify how many models would be needed to obtain a "quasi-final" PLM.

The findings are: (i) cross table analysis used to determine exclude relationships; (ii) association rules analysis help identified mandatory and optional relationships; (iii) chi-square independence test combined with association rules are an effective way to identify require relationships; (iv) while constructing the PLM, errors are detected in PMs. The overall PLM construction process should thus be iterative and intertwined with PM correction; and (v) the techniques are efficient enough to be applied even on a large collection of PMs.

The rest of the paper is structured as follows. The next section presents mining techniques that we considered while developing the method. Section 3 presents our method and reports the rationale for the technique actually used at each step. Section 4 reports our evaluation in a real case. The concluding section presents related and future works.

II. MINING TECHNIQUES

Some mining techniques can be used in order to find relationships among a collection of variables. The better adapted to discover constraints between features are:

A. Cross Table Analysis

The cross table analysis consists in a paired based comparison among the different features. Normally, it is represented as a $n \times n$ matrix that provides the number of co-occurrences or conditional probabilities between features.

B. Association Rules

The objective of association rule mining [6] is the elicitation of interesting rules from which knowledge can be derived. Those rules describe novel, significant, unexpected, nontrivial and even actionable relationships between different features or attributes [7], [8]. Association rule mining is commonly stated as follows [9]: Let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of *items*, and D be a set of transactions. Each transaction consists of a subset of items in I. An association rule, is defined as an implication of the form $X \rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \phi$. Support, confidence, Chi square statistic and the minimum improvement constraint among others might be considered as measures to assess the

quality of the extracted rules [10]. Support determines how often a rule is applicable to a given data set of an attribute and it represents the probability that a transaction contains the rule. The confidence of a rule $X \rightarrow Y$ represents the probability that Y occurs if X have already occurred $P(Y/X)$; then it estimates how frequently items Y appear in transactions that contain X. Chi square statistics combined with its test (see next section) might be used as a measure to estimate the importance or strength of a rule from a given set of transactions and by this way to reduce the number of rules [11]. Finally the minimum improvement constraint measure not only indicates the strength of a rule but it prunes any rule that does not offer a significant predictive advantage over its proper sub-rules [12].

In this work, in the process for obtaining rules, we consider the Apriori Algorithm [9] that is supported on frequent item sets and is based on the following principle:

"*If an itemset is frequent, then all of it subsets must also be frequent" Conversely "If an item set is infrequent, then all of it supersets must be infrequent to*".

For the purpose of this work items will be considered as features and transaction as PMs and the result of this pair wise is what we call the binary features matrix.

C. Chi Square and Independence Test

This test is based on Chi square value measure [11]. The measure is obtained by comparing the observed and expected frequencies, and using the following formula:

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \qquad (1),$$

where O_i stands for observed frequencies, E_i stands for expected frequencies, and i runs from $1,2,\dots,n$, where n is the number of cells in the contingency table.

The value obtained in equation 1 is then compared with an appropriated critical value of Chi square. This critical value chi-square χ_0^2 depends of the degrees of freedom and level of significance. The critical value chi-square χ_0^2 will be calculated with $n-1$ degrees of freedom and α significance level. In other words, when the marginal totals of a 2 x 2 contingency table is given, only one cell in the body of the table can be filled arbitrarily. This fact is expressed by saying that a 2 x 2 contingency table has only one degree of freedom. The level of significance α means that when we draw a conclusion, we may be $(1-\alpha)$ % confident that we have drawn the correct conclusion (normally the α value is equal to 0.05). For 1 degree of freedom and a significance level of 0.05 critical value chi-square χ_0^2 =3.84.

The most common use of the test is to assess the probability of association or independence of facts. It consists on testing the following hypothesis:

Hypothesis null H_0: The variables are independent.

Alternative hypothesis H_1: The variables are NOT independent.

In every chi-square test the calculated χ^2 value will either be (i) less than or equal to the critical χ_0^2 value OR (ii) greater that the critical χ_0^2 value. If calculated $\chi^2 \leq \chi_0^2$ we conclude that there is sufficient evidence to say that cross categories are independent; otherwise can think on dependency.

III. PROPOSED METHOD

A. Method Overview

The main concerns in PLM construction are:

1) Preparation: to begin with our approach it is necessary to dispose of a collection of related features or artefacts for each application. Artefacts or features could be extracted from repositories and by means of clustering process the hierarchical relation could be established [3].

In another hand, a part generalization is required. Text mining techniques are used to deal with this generalization. In fact i.e Romanowski in [13] uses a neural network based text analysis program to generalize parts.

2) *Structural Dependency Identification*: to determine parent child relationships and also characterize which of them are mandatory and or optional;

3) Transversal Dependency Specification: to study the behavior among features that are not member of the parent child link and exploit not only all the possible mutual exclusive relationships known as "excludes", but also distinguish all the relationships that indicate where a specific features may "require" the selection of another feature.

4) Grouped Cardinality Specification. Optional features that have the same father can be bundled, and constraints can be specified to indicate how many (at most and at least) features of the bundle can be selected together in a single product;

5) *Consolidation*. Results from previous concerns are evaluated by an expert.

Each of the following sections explain in which mining technique is proposed to support each of these phases

B. Preparation

Our approach is based on constructing a product line model based on existent product models. Then, to consider our approach and to successfully implement it, it is strictly required to get a collection of product models or related artifacts or features.

In order to execute our approach, we need a set of refinement relationships between features, that is, child-father tuples in two forms a list of relationships and its derivate matrix of feature occurrence in PM. This matrix is obtained by highlighting the features presence in product models.

P1 , F1
F1 , F5
F1 , F6
P2 , F1
F1 , F6
F1 , F4
F1 , F5

a)

	F_1	F_2	F_3	F_4	F_5	F_6	F_7
P_1	1	0	0	0	1	1	0
P_2	1	0	0	1	1	1	0

b)

Fig. 1. a) List of relationships. Column left represents fathers and right their childs. P_1 is father of F_1, F_1 father of F_5 and so on. b) PM -Feature binary matrix. The feature takes the value 1 if it is present in a product model and zero otherwise.. For instance F_1, F_4, F_5, and F_6 are present in P_2 and contrary F_2 and F_7 are never taken into consideration by any product model.

C. Structural Analysis

From a collection of product models and their structure it is possible to determine i) bundles, parents and children; ii) the feature binary matrix, input for the association rule mining. This step is handled by means of the Apriori algorithm to then obtain the mandatory and optional relationships.

Step 1: Identifying Structural Patterns. Due to the nature of input the identification of parents, sons and bundles consists on browsing this relational structure of a product model. The most representative example is the bill of material in manufactured finish good. The composition of the finish good is represented as a relational table that mainly integers the parent item and its components or children

Step 2: *Running Association Rules Apriori Algorithm*. Once the binary feature matrix is built, we have the input to apply the association rule data mining tool, that permit us not only to explore the relationships and dependencies but also to handle a huge amount of data in an optimal way. However, such algorithms developed are sometimes limited to the memory because of its size and calculus that they perform.

In fact the most complex task of the whole association rule mining process is the generation of frequent itemsets (in this part an itemset is considered as feature set). Many different combinations of features and rules have to be explored which can be a very computation-intensive task, especially in large databases. By setting the parameter association rule length equals to 1 for the Apriori algorithm, we can study only singles relations between features to avoid those computation complexities. Often, a compromise has to be made between discovering more complex rules and computation time.

To filter those rules that might be not valuable, it is important to calculate its support. As we have already seen, the support determines how frequent the rule is applicable to the product P. This value compared with the minimum support accepted by an expert (min support threshold), prunes the uninteresting rules.

To evaluate the interestingness and pertinence; that is it the reliability of the inference made by a rule, it is useful to evaluate its confidence. The task is now to generate all possible rules in the frequent feature set and then compare their confidence value with the minimum confidence (which is again defined by the expert). All rules that meet this requirement are regarded as interesting. All the final discovered associations with their support and confidence values, therefore, may be presented to stakeholders.

Furthermore the calculation of other measures is relevant to refine the process of selecting the appropriate association rule. For that we propose to calculate the Chi-square and to indicate the strength of a rule. The minimum improvement constraint measure not only gives us an idea about the strength but also prunes any rule that does not offer a significant predictive advantage over its proper sub-rules. This increases efficiency of the algorithm, but more importantly, it presents the user with a concise set of predictive rules.

Step 3: Identifying Mandatory Relationships Using Association Rules. Removing all association rules that do not satisfy the minimum improvement constraint, offers us the most relevant and significant rules available for the study.

It is obvious that those relationships that are always present in all the product models may be considered as mandatory. Now, if some ambiguous information is present in the database and this one is not reliable at λ%, in order to obtain mandatory relationships, the analyst may establish as a minimum confidence threshold the value (100- λ)%. Those rules whose confidence is greater than the (100- λ)% may be considered as mandatory relationships. Bidirectional rules such as $F1 \rightarrow F2$ and $F2 \rightarrow F1$ may be also considered as mandatory relationships [14].

The relationship is classified as mandatory if at least one of the two properties mentioned before (high frequent features and bidirectional rules) occurs and, of course, the relationships belong to a parent child.

Step 4 Identifying Optional Relationships. Once parent child and as well mandatory relationships are identified the remaining parent child relationship may be classified as optional.

D. Transversal Analysis

By combining some results obtained from the previous sections such as the PM feature binary matrix and parental relationships with a cross tabulation analysis among features and an independence test to detect strong relationships, it is possible to identify exclude and requires relationships.

Step1: Identifying Exclude Relationships. Feature cross table display relationships between features. Let $F = \{F_1, F_2, \dots, F_n\}$ be a set of n features. $F \times F$ can be represented as a $n \times n$ cross table describing the joint occurrence between the feature i and j. When the joint distribution of (F_i, F_j) for all $i \neq j$ is equal to zero, that can be interpreted that there is no probability that F_i and F_j may occur at the same time. Thus, they are mutually exclusive and the relationship between F_i and F_j is considered as an exclude relationship.

A further analysis of contingency table could give us valuable information about some types of relationships such as mandatory, optional and requires.

Step2: Identifying Requires Relationships To identify requires relationships it is necessary to apply a Chi-square independence test. The test is performed for each single rule with 1 degree of freedom in order to prove with a significance level α =0,05 that the relationships between non parent-child features F_i, F_j for all $i \neq j$ are independent or not.

Thus, the association between F_i, F_j for all $i \neq j$ is considered as *dependent* if the χ^2 value for the rule with respect to the whole data exceeds the critical χ^2 =3.84 (χ^2 critical value with one degree of freedom and a significance level α =0,05) otherwise it is considered as *independent*.

E. Grouped Cardinality Analysis

This process helps the analyst in assigning the group cardinality value. It is interesting for the analyst to have a tool

that allows him to estimate the cardinality for each non mandatory optional bundle.

Step1: Identifying All Possible Feature Sets for Each Bundle. All the possible optional features sets in each bundle are captured by browsing the product line model structure.

Step2: Counting feature's occurrences for each product model and optional bundle set obtained in step 1.Here we evaluate each PM and display how many features from the group are present in the configuration. As a feature in our work is considered as a binary variable, by examining the presence of the group and the related features related in each product model, it is possible to obtain the group occurrence by adding their respective feature values.

IV. STUDY CASE

Our method was validated with the construction of the Baxter Bioscience Lessines product line model. Baxter Biocience at Lessines-Belgium develops, manufactures and markets products for hemophilia and immune disorders.

To construct the packaging product line model, we focused our study around all the components that constitute the packaging process of the different treatments that Baxter Bioscience produces. We have worked with 536 packaging bill of materials (BOM) as product models and we have also handled more than 1500 items. After generalizing items, we proceed to apply our approach and evaluate the results obtained by estimating the algorithm time complexity and the scalability generating the desired constraints. First, examining the time complexity of the algorithm that supports our approach, we have observed that it is really efficient but it presents some limitations when studying group cardinalities. Second, performing a paired comparison of constraints generated from different random products samples. We can observe structural dependencies show a high predictive capacity: 95% of the mandatory and optional relationships are founded when we take a random sample size of at least 350 products. The totality of the mandatory relationships are then discovered when the random sample size is greater than 450 products, however excludes and, especially, requires relationships, seem to depend to the problem size that is it, the number of constraints increases when sample size increases. This can be explained by examining the nature of the data used in our study case. Structural relationships mainly depend on the composition of the product; thus they depend of the parent child relationships or BOM composition and transversal dependencies are related to relationships attributes. More products means more attributes, and at the end, this means that more transversal relationships to be discovered.

V. CONCLUSIONS AND FUTURE WORKS

Our work is one of the first real scale experience of automation of the construction process of PLMs. To our knowledge, it is on of the first approaches that integrates statistical techniques to identify commonalities and variabilities in a collection of a non predefined number of product models. Indeed, although rigorous, our proposal needs to be expanded and benchmarked with respect to alternative strategies explored, and implemented into a marketable tool.

Our experience showed that there is a need for a method that is able to deal with richer input information. For example, we had products that are defined with more complex than Boolean-type features, as for instance scalar variable (*e.g*. integer or real values as in performance characteristics of systems) or set variables (when system features can be instantiated a varying number of times in the same products). As a consequence, we believe that more complex relationships can be needed in the target PL models. How can these be specified? Remain still an open question for future researches.

Several other fundamental questions are still open and their solutions are envisaged for future works. For instance: what is a good quality model to construct a product line model? How to deal with ambiguous information to construct a product line model? How to deal with more complex constraints? What statistical tools could be used to support the aforementioned questions?

REFERENCES

[1] Marinelli F., de Weck O. ,Krob D. , Liberti L., A General Framework for Combined Module- and Scale-based Product Platform Design, Second Internl Symp on Engineering Systems MIT, Cambridge, Mass, 2009.

[2] Streitferdt, D.: FORE Family-Oriented Requirements Engineering, PhD Thesis, Technical University Ilmenau, 2004.

[3] Chen K, Zhang W, Zhao H, Mei H. An Approach to Constructing Feature Models Based on Requirements Clustering. Internl Conf on Req Eng. pp 31-40 Paris 2005.

[4] Moon S K., Kumara S R T., Simpson T W. Data mining and fuzzy clustering to support product family design, Proc of IDETC/CIE 2006.

[5] Al-Otaiby T N, Alsherif M, Bond W. Towards Software Requirements Modularization using Hierarchical Clustering Techniques. 43rd Southeast regional Conference, Vol 2, Georgia, pp 223-228, 2005.

[6] Ceglar,J.F Roddick. Association mining. ACM Computing Surveys (CSUR), Vol 38 Issue 2, 2006.

[7] Agar, B. and Kusiak, A, "Data-mining-based Methodology for the Design of Product Family," *International Journal of Production Research*, vol. 42, No. 15, pp. 2955-2969, 2004.

[8] Jiao, J. and Zhang, Y., "Product Portfolio identification based on Association Rule Mining," *Comp.-Aided Design*, vol. 27, No. 149-172,,2005

[9] Agrawal, R., Imielinski, T., Swami, A. "Mining association rules between sets of items in large databases." SIGMOD-1993, pp 207-216, 1993

[10]Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to data mining. Ch 6: Association Analysis: Basic Concepts and Algorithms. Addison Wesley 2006

[11]B Liu, W Hsu, Y Ma. Pruning and Summarizing the Discovered Associations. In ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-99), August 15-18, , San Diego, CA, USA, 1999

[12]Bayardo, R. J.; Agrawal, R.; and Gunopulos, D. Constraint-Based Rule Mining in Large, Dense Databases. In Proc. of the 15th Int'l Conf. on Data Engineering, pp 188-197. 1999.

[13]Romanowski, C.J. and Nagi, R., A data mining approach to forming generic bills of materials in support of variant design activities, ASME J of Computing and Information Science in Eng., 4(4), 316–328,2004

[14]Batory, D.; Thaker, S. Towards Safe Composition of Product-Lines. Dept. Computer Sciences, University of Texas, TR-06- 33, 2006.