Context-Aware User and Service Profiling by Means of Generalized Association Rules*

Elena Baralis[1], Luca Cagliero[1], Tania Cerquitelli[1], Paolo Garza[1], and Marco Marchetti[2]

[1] Politecnico di Torino - Dipartimento di Automatica e Informatica - Torino, Italy
{elena.baralis,luca.cagliero,tania.cerquitelli,paolo.garza}@polito.it
[2] Telecom Italia Lab - Torino, Italy
marco1.marchetti@telecomitalia.it

Abstract. Context-aware applications allow service providers to adapt their services to actual user needs, by offering them personalized services depending on their current application context. Hence, service providers are usually interested in profiling users both to increase client satisfaction, and to broaden the set of offered services.

Since association rule extraction allows the identification of hidden correlations among data, its application in context-aware platforms is very attractive. However, traditional association rule extraction, driven by support and confidence constraints, may entail either (i) generating an unmanageable number of rules in case of low support thresholds, or (ii) discarding rare (infrequent) rules, even if their hidden knowledge might be relevant to the service provider. Novel approaches are needed to effectively manage different data granularities during the mining activity.

This paper presents the CAS-Mine framework to efficiently discover relevant relationships between user context data and currently asked services for both user and service profiling. CAS-Mine exploits a novel and efficient algorithm to extract generalized association rules. Support driven opportunistic aggregation is exploited to exclusively generalize infrequent rules. User-provided taxonomies on different attributes (e.g., a geographic hierarchy on spatial coordinates, a temporal hierarchy, a classification of provided services), drive the rule generalization process that prevents discarding relevant but infrequent knowledge.

Experiments performed on both real and synthetic datasets show the effectiveness and the efficiency of the proposed framework in mining different types of correlations between user habits and provided services.

Keywords: Generalized association rules, knowledge discovery, context-aware data.

1 Introduction

Context-aware systems acquire and exploit information on the user context to tailor services to the particular user, place, time, and/or event. Research activities

* This work was supported by a grant from Telecom Italia Lab.

J.D. Velásquez et al. (Eds.): KES 2009, Part II, LNAI 5712, pp. 50–57, 2009.

on context-aware computing have been devoted both to exploring the different dimensions of context-awareness [8], and to implementing different context-aware applications (e.g., in the medical domain [15], for mobile phones [6]). Context could consist of any circumstantial factors or application context users are involved in. Thus, context-awareness means that the system is able to exploit context information. A system is context-aware if it can extract, interpret and use context information and adapt its functionalities to the current usage context [9]. Provided services could be personalized by exploiting either the current context of the user [4] or historic context and behavior of the user [5]. An in-depth literature review on context-aware systems has been presented in [8].

Authors in [16] first proposed to exploit statistical machine learning techniques (e.g., rule induction, neural networks, Bayesian networks) to build predictive user models. These models have been exploited in different context-aware settings (e.g., a smart home [12], or a smart office [11]). Application and user profiling, instead, have been addressed in [10]. The proposed solution exploits rule based and naive Bayesian classifiers. In particular, different service and application models are tailored to the user and to the situation in which she is involved. These models are then exploited to suggest applications and services on what the user might interest in her current situation.

Association rule extraction [1] is a widely used exploratory technique allowing the discovery of hidden correlations among data. Its application in context-aware platforms to profile both users and services is very attractive. However, traditional association rule algorithms are not effective in mining context datasets because of the high detail level of the information (e.g., GPS coordinates). When low support thresholds are enforced, an unmanageable number of rules is extracted, while high support thresholds discard rare rules even if their knowledge might be relevant for the service provider. To address this issue, rules should be extracted at a higher abstraction level (i.e., generalized).

This paper presents the CAS-Mine framework to efficiently discover relevant relationships between user context data and currently requested services. Extracted rules may be exploited both for user and service profiling. To this aim, CAS-Mine exploits a novel and efficient algorithm, called GenIO, to extract generalized association rules. The GenIO algorithm extends the concept of multi-level rules [7] by performing an opportunistic extraction of generalized itemsets. It exploits (user provided) taxonomies to drive the itemset generalization process and efficiently extract generalized itemsets. Instead of extracting itemsets for all levels of the taxonomy and post-pruning them [7], the generalization step over the taxonomy is support driven, i.e., it generalizes an item climbing up the taxonomy if and only if its support is below the support threshold. The generalization process stops when the newly generalized item is above the support threshold.

Preliminary experiments performed on both real and synthetic datasets show the effectiveness and efficiency of the CAS-Mine framework in highlighting interesting rules to characterize users and services.

The paper is organized as follows. Section 2 motivates our work. Section 3 presents an overview of the CAS-Mine framework and describes its main features. In Section 4 preliminary experiments to validate the proposed framework are reported, while Section 5 draws conclusions and discusses future work.

2 Motivations

A structured context dataset holds information on service requests performed by users and the corresponding application context. Each data element is a set of items describing a service request and its context. Each item is a couple *(AttributeName,Value)*. The *AttributeName* describes the represented information (e.g., *user identifier*, *service*, *time*), while *Value* is the actual value of the corresponding attribute (e.g., *ID54*, *weather*, *4:06pm*). A generalized item is defined by means of a user-defined hierarchy of aggregation (i.e., taxonomy) over values in the attribute domain. For example, the *position* attribute may describe GPS coordinates (e.g., 45.438:12.335). The couple *(position, 45.438:12.335)* is an item (at the lowest level in the hierachy), while the couple *(position, office)* is a generalized item. Thus, *office* aggregates all GPS coordinates related to the office physical location.

Generalized association rules are represented in the form $A \Rightarrow B$ (s%,c%), where A and B are sets of (possibly generalized) items, and s% and c% represent support and confidence. The support is the prior probability of A and B (i.e., its observed frequency in the dataset). The confidence is the conditional probability of B given A and characterizes the "strength" of a rule. User activity may be characterized by the following association rule

`user`: John, `time`: 6.05 p.m. $\Rightarrow$ `service`: Weather ($s = 0.005\%, c = 98\%$)

This specific rule is characterized by a very low support and is not extracted, because the extraction process would become unfeasible. By generalizing the time attibute on a time period, and the user on a user category, the following generalized rule may be obtained.

`user`: employee, `time`: 6 p.m. to 7 p.m. $\Rightarrow$ `service`: Weather ($s = 0.2\%, c = 75\%$)

If the obtained rule is still below the support threshold, the generalization process performs a further aggregation step on the time hierarchy.

`user`: employee, `time`: Evening $\Rightarrow$ `service`: Weather ($s = 1.5\%, c = 65\%$)

The generalization process greedily continues until the obtained rule is above the selected support threshold. Thus, generalization allows highlighting interesting correlations which would be lost because of their low support at the lowest level of the hierachy.

3 The CAS-Mine Framework

CAS-Mine is a framework to efficiently profile both users and services. Thus, it allows shaping service supply by considering the context to which the user

belongs. By discovering recurrent patterns involving user habits and requested services, providers may partition users into a set of well-known categories for which supplied services may be modeled and personalized.

CAS-Mine exploits the GenIO algorithm, a novel and efficient algorithm to discover generalized association rules. The mining process is driven by a set of (user-provided) taxonomies which allow the conceptual aggregation of items in more abstract categories. The main blocks of the CAS-Mine framework are reported in Figure 1, while a more detailed description of each block is presented in the following.

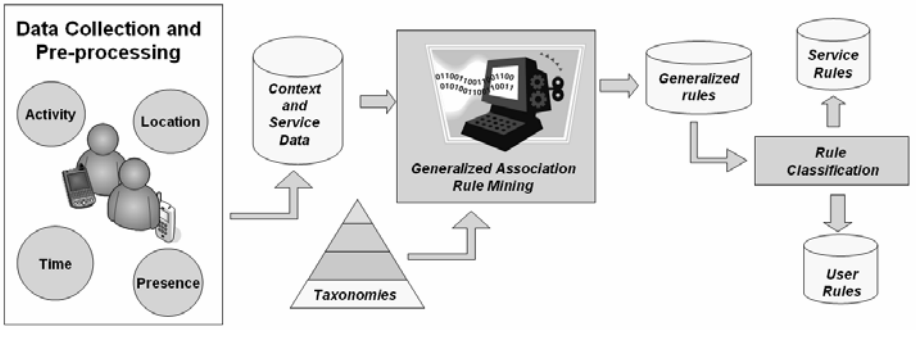

Fig. 1. The CAS-Mine Framework Architecture

3.1 Data Collection and Pre-processing

The data collection and pre-processing block manages data collection from a large number of mobile devices which provide information on the user context and on the supplied services (e.g., temporal information, GPS coordinates, service description).

The data collection block receives in input the raw context data provided by different, possibly heterogeneous, sources and integrates them into a common data structure. During this process, irrelevant and redundant information is also removed. Cleaned and integrated data are finally stored into a common repository.

3.2 Generalized Association Rule Mining

This block performs the extraction of generalized association rules. Extraction is performed in two steps: (i) frequent generalized itemset extraction and (ii) rule generation from the extracted frequent itemsets. Since itemset mining is the most computationally intensive step [1], the novel contribution of the GenIO algorithm focuses on itemset mining. The second step exploits Goethal's Rules software [3], possibly enforcing confidence constraint.

Given a dataset, a set of user-provided multi-level taxonomies (at most one for each attribute) and a minimum support threshold, the GenIO algorithm extracts all the frequent not-generalized itemsets and the set of generalized

itemsets which represent the generalization of the knowledge associated to infrequent not-generalized itemsets.

GenIO performs a support-driven opportunistic aggregation of infrequent itemsets, thus avoiding exhaustive multi-level extraction followed by post-pruning. Hence, the GenIO algorithm reduces the cardinality of mined itemsets with respect to well-known traditional multi-level algorithms [7]. GenIO successfully tackles both excessive pruning and computationally hard exhaustive multi-level extraction, thus providing a good trade-off between itemset specialization and aggregation.

The itemset mining process, driven by the generalization step, is shortly described in the following. A detailed description of the GenIO algorithm is provided in [2]. GenIO is an Apriori-like [1] extraction algorithm, which performs level-wise itemset mining. In general, Apriori-like algorithms, at a generic level i, extract all the frequent itemsets of length i. Candidate itemsets of level $i+1$ are generated by combining all the frequent itemsets of length i. Frequent itemsets of level $i+1$ are then obtained by enforcing the support constraint.

The GenIO algorithm exploits rule generalization to extract knowledge on infrequent, but possibly interesting, itemsets. At each extraction level, before pruning itemsets not satisfying the support threshold, generalizes them by climbing up the generalization hierachy of the corresponding items. A generalized entry is a conceptual aggregation of different items at a lower level (e.g., *Communication service* represents lower level items *SMS service* and *CALL service*), thus it is more likely to be frequent. Taxonomies are exploited to drive the generalization process (see Section 2). GenIO climbs up each taxonomy in a stepwise fashion, until either support constraint is satisfied, or the highest generalization level is reached.

3.3 Rule Classification

The rule classification block categorizes generated rules in classes to effectively exploit them for different context-aware profiling. Since service providers are tipically interested in profiling both users and services, the CAS-Mine framework currently identifies two classes of association rules: (i) User rules and (ii) service rules. User rules characterize user habits at any aggregation level. These rules allow service providers to offer personalized services tailored to the current context of the user. Hence, provided services can be adapted to actual user needs. Service rules describe service characteristics, at any hierarchical level, without specific user information. These rules allow service providers to adapt service provisioning to the current context, independently of the requesting user (e.g., by providing a different bandwidth in different time periods).

4 Preliminary Experimental Results

We evaluated the CAS-Mine framework by analyzing (i) the characteristics and interestingness of extracted patterns on a real dataset, and (ii) the scalability, in terms of execution time, of the proposed approach on a synthetic dataset. All

experiments were performed on a 3.2-GHz Pentium IV system with 2 GB RAM, running Ubuntu 8.04.

4.1 Characterization of Extracted Rules

The real dataset, denoted as *mDesktop*, has been provided by Telecom Italia Lab (Tilab). The trial version of the Tilab mobile desktop application provides different services to users (e.g., weather forecast) on mobile devices. The mDesktop dataset is characterized by 4487 records with information on each requested service and the context of the requesting user (e.g., time, location). The dataset is characterized by the following taxonomies:

- date → month → year
- timestamp → hour → day period (AM/PM)
- service → class of service
- latitude:longitude → city → country
- phone number → call type (PERSONAL/BUSINESS)

Figure 2 reports statistics about the mining activity performed on the mDesktop dataset. By setting high minimum support thresholds (e.g., 10%), only very frequent rules are extracted. In particular, only the generalized rules composed by the top levels of the taxonomies are extracted (e.g., `location`: ITALY ⇒ `date`: 2008). These rules are usually too general to provide interesting knowledge. Differently, when lower threholds are enforced, the extracted generalized rules include non-top level elements of the taxomonies, which may provide more actionable knowledge.

In the following, we analyze two different rule subsets, which show the effectiveness of the GenIO algorithm in supporting both user and service profiling.

User profiling rules. These rules deal with context-aware profiling for specific users whose habits show some kind of recurrence. In particular, for an arbitrary user, the GenIO algorithm highlights the service type the user is mainly interested in, the context in which requests are commonly submitted, and the service parameters. The following two rules (support threshold=1%, absolute threshold=45) discover valuable knowledge about an anonymous client, denoted here as *Rossi*:

A) `user`: Rossi ⇒ `service`: CALL ($s = 1.27\%, c = 53\%$)
B) `user`: Rossi ⇒ `service`: SMS ($s = 1.14\%, c = 47\%$)

The above rules highlight that user *Rossi* is interested in two specific services, CALL and SMS, with confidence close to 50%. Thus, they provide a relevant knowledge on this user preferences. When the (higher) support threshold 2% is enforced (absolute threshold = 90), the following generalized rules are extracted.

A) `user`: Rossi ⇒ `hour`: PM ($s = 2.25\%, c = 94\%$)
B) `user`: Rossi ⇒ `service`: Communication ($s = 2.41\%, c = 100\%$)

The first rule shows an intensive system usage for user *Rossi*, especially during the afternoon/evening (confidence 94%). Furthermore, the second rule, with confidence 100%, shows that *Rossi* is exclusively interested in the *Communication* service class, which contains the CALL and SMS services. A traditional mining

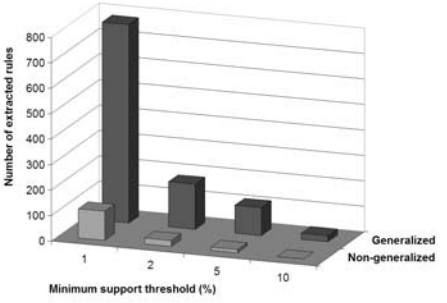

Fig. 2. Rule statistics

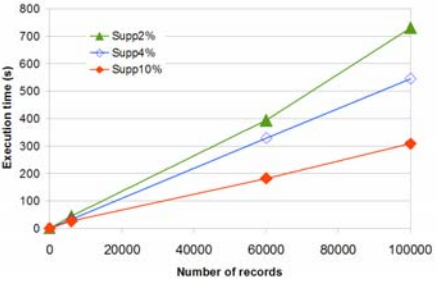

Fig. 3. Scalability on TPC-H datasets

algorithm would lose the above knowledge by pruning the lower level infrequent itemsets. GenIO capability of automatically climbing up the service taxonomy allows the extraction of these interesting high level rules.

Service profiling rules. These rules characterize frequently used services, independently of the specific user requesting them. The following generalized rules are extracted with support threshold 1% (absolute threshold = 45).
A) `date`: August, `hour`: PM ⇒ `service`: HOME ($s = 7.09\%, c = 94\%$)
B) `service`: CALL ⇒ `inout`: OUT ($s = 1.14\%, c = 89\%$)
The first rule shows that 94% of service requests submitted in the evening during the month of August are related to the HOME service. This knowledge may be exploited both to size system resources, hence providing a more efficient HOME service, and to select the first service to suggest to connected users. The second rule highlights the correlation between the service type and its parameters. In this case, call services are mainly exploited to perform outgoing calls.

4.2 Scalability

We analyzed the scalability of the GenIO algorithm with respect to the cardinality of transactions on syntethic datasets generated by means of the TPC-H generator [13]. By varying the scale factor parameter, tables with different cardinalities are generated. We generated datasets of size ranging from 6,000 to 100,000 transactions with 8 categorical attributes. We mined generalized itemsets from the *lineitem* table and we exploited the *part*, *nation*, and *region* tables to define taxonomies on line items.

Figure 3, which plots the extraction time for various supports, shows that the proposed algorithm scales well also for large datasets. Since the number of extracted itemsets grows for low supports (e.g., 2%), the process becomes computationally more expensive. However, the overall CPU time is still low, less than 730s for the lowest considered support and largest dataset.

5 Conclusion and Future Works

Context-aware applications exploit implicit context information (e.g., environmental conditions, location, time) to enhance the fulfillment of explicit user

requests by providing personalized services tailored on the current application context of the user. In this paper we presented the CAS-Mine framework to support context-aware user and service profiling. CAS-Mine exploits a novel algorithm to efficiently mine generalized association rules. The mining process is driven by user-provided taxonomies on different attributes, which prevent discarding relevant but infrequent knowledge.

Future extensions of the framework will address (i) the automatic inference of taxonomies from the input context dataset, (ii) the exploitation of multiple taxonomies over a single attribute, and (iii) the application of the opportunistic generalization approach to more efficient rule extraction algorithms (e.g., LCM [14]).

References

1. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules in Large Databases. In: Proceedings of the 20th VLDB conference, pp. 487–499 (1994)
2. Baralis, E., Cerquitelli, T., D'Elia, V.: Generalized itemset discovery by means of opportunistic aggregation. Technical report, Politecnico di Torino (2009), `https://dbdmg.polito.it/twiki/bin/view/Public/NetworkTrafficAnalysis`
3. Goethals, B.: Frequent Pattern Mining Implementations, `http://www.adrem.ua.ac.be/~goethals/software`
4. Bradley, N.A., Dunlop, M.D.: Toward a multidisciplinary model of context to support context-aware computing. Hum.-Comput. Interact. 20(4), 403–446 (2005)
5. Byun, H., Cheverst, K.: Utilizing context history to provide dynamic adaptations. Applied Artificial Intelligence 18(6), 533–548 (2004)
6. Hakkila, J., Mantyjarvi, J.: Collaboration in context-aware mobile phone applications. In: Hawaii International Conference on System Sciences, vol. 1, p. 33 (2005)
7. Han, J., Fu, Y.: Mining multiple-level association rules in large databases. IEEE Trans. Knowl. Data Eng. 11(5), 798–804 (1999)
8. Hong, J., Suh, E., Kim, S.: Context-aware systems: A literature review and classification. In: Expert Systems with Applications (November 2008)
9. Khedo, K.K.: Context-aware systems for mobile and ubiquitous networks. In: ICNICONSMCL 2006, p. 123 (2006)
10. Nurmi, P., Salden, A., Lau, S.L., Suomela, J., Sutterer, M., Millerat, J., Martin, M., Lagerspetz, E., Poortinga, R.: A system for context-dependent user modeling. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2006 Workshops. LNCS, vol. 4278, pp. 1894–1903. Springer, Heidelberg (2006)
11. Oliver, N., Garg, A., Horvitz, E.: Layered representations for learning and inferring office activity from multiple sensory channels. Comput. Vis. Image Underst. 96(2), 163–180 (2004)
12. Tapia, E.M., Intille, S.S., Larson, K.: Activity recognition in the home using simple and ubiquitous sensors. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 158–175. Springer, Heidelberg (2004)
13. TPC-H. The TPC benchmark H. Transaction Processing Performance Council (2009), `http://www.tpc.org/tpch/default.asp`
14. Uno, T., Kiyomi, M., Arimura, H.: LCM ver. 2: Efficient mining algorithms for frequent/closed/maximal itemsets. In: FIMI (2004)
15. Vajirkar, P., Singh, S., Lee, Y.: Context-aware data mining framework for wireless medical application. In: Mařík, V., Štěpánková, O., Retschitzegger, W. (eds.) DEXA 2003. LNCS, vol. 2736, pp. 381–391. Springer, Heidelberg (2003)
16. Zukerman, I., Albrecht, D.W.: Predictive statistical models for user modeling. User Modeling and User-Adapted Interaction 11(1-2), 5–18 (2001)