Analyzing Service Order Data using Association Mining

Ayhan Demiriz
Information Technology, Verizon Inc.,
919 Hidden Ridge, Irving, TX 75038
E-mail:ayhan.demiriz@verizon.com

Abstract
Association mining was originally developed and successfully applied to analyze the sale transactions of retail goods. According to common practices of association mining, each item is treated equally. But in a service order transaction - especially in a telecom setup, an item (service) can be in several states. We propose a simple solution to analyze service order data using association mining by tagging items with corresponding state identifiers. We successfully apply our result to an association mining based product recommender system. Results from random and naïve product recommender systems are also reported for the comparison purposes.
Keywords: Association Mining, Service Order, Product Recommendation, e-commerce

1 Introduction

In this paper, we introduce a very simple but efficient as well as sufficient way of analyzing service order data - specially data from a telecom environment. We also show that results can be applied to product recommendation. This study has been conducted to show the feasibility of the approach. Proposed system is not in production yet.

The main purpose of association mining is to analyze market basket data. In this case each item treated equally and association mining is conducted to find association rules among these items. The resulting rules are represented in an 'if' statement framework such as: $if\ A\ \&\ C \rightarrow B$. The rule reads usually (depending on the context) **if item A and C are purchased together then item B is also purchased with confidence level x% and support level y%**. In a common usage of association mining, items can be discounted and sold accordingly in supermarkets based on found rules. But one point might be overlooked by data analysts inadvertently. Certain items can heavily be discounted and campaigned together with other items. Such sale events might cause very skewed item frequencies which would result in misleading rules. Because in reality, an association might be valid if one or more of the items are on sale (discounted). To reflect this reality and represent the different states that an item can be in, data analysts should distinguish the discounted items from regular ones.

To overcome this problem, we simply propose tagging the items accordingly. In the scenario mentioned above, discounted items can be tagged by adding 'D_' in front of the item name. Similarly, service order transactions are composed of items in several states. A service can be

- Added as a new one
- Kept (retained) without any change
- Removed (deleted) from the customer profile
- Used (changed) as part of a new product or service.

From a product recommendation perspective, finding associations among products (services) with their corresponding states might be potentially useful to evaluate the current marketing strategies and to recommend better solutions to the prospective and current customers.

The purpose of this paper is to show that our approach is feasible to implement and can result in very practical outcomes. The remaining of this paper is organized as follows. We introduce problem domain and give a short description of the proposed system in Section 2. We then explain the algorithms used for product recommendation in Section 3. Experimental results are discussed in Section 4. Related work briefly discussed in Section 5. The paper ends with possible extensions to the current work and concluding remarks.

2 Problem Statement and System Description

To stay competitive and maintain high level customer satisfaction service companies usually offer new or improved services based on the existing services. In such cases, customers are encouraged and targeted to switch to these new services. This is true for existing customers and this type of service order is considered as a change order. For the new customers and leaving customers orders are treated separately. Proposed system in this paper is designed for only change orders.

Service upgrades (or downgrades) usually substitute the current services. On the other hand new services might require additional products and services. In a such scenario, they complement certain services and products. The effects of substitute and complement products are called 'Cross Elasticity of Demand' in marketing and economics literature. Our aim in this study is to find associations among services at the time of service orders to understand the customer preferences and to monitor the sales. As summarized before a service (product) can be added, retained, changed or deleted in a given service order. Thus state information is also used in finding associations by appropriate tagging mentioned above. For this purpose, identifiers A_, R_, C_, and D_ are added to the beginning of the service names respectively. In telecom industry service names are called as Universal Service Order Code (USOC). We conducted this study at USOC level. Proposed system is designed by considering only those service orders generated in Call Centers for Residential Customers but can be generalize to the other sales channels.

An overview of steps in this study given below. For the first step, activities of approximately 200 Customer Service Representatives (CSR) -starting with a few-have been observed since June 2002. All service orders generated by these CSRs are tracked and populated in realtime from back-end systems to a local database. As mentioned above, only change orders are considered in this study.

- Data collection and preparation.
- Association mining on tagged service names.
- Screening association rules
- Given retained, deleted, and changed products, we predict added product and make recommendation based on screened association mining rules.

Certain change orders might have only retained and deleted services. From a product recommendation perspective, at least one added product or service is needed. Thus change orders are further screened to include only those change orders that have at least one added service.

To simulate the real prediction case, we use the data collected between June 11 and September 17 as our training data and data collected between September 17 and September 23 as test data. We prefer using this strategy to reflect problems faced in a live application.

Table 1: Summary of the Datasets

	Dataset	
Statistics	Training	Test
Number of USOCs	260	201
Number of Tagged USOCs	688	472
Number of Transactions	76781	27772
Ave. Number of Items	7.69	7.42
Ave. Number of Added Items	1.95	1.51
Number of Nonzero Entries	590134	206131
Sparsity	0.9888	0.9843

We summarize characteristics of both training and test data in Table 1. The sparsity of each dataset is defined as $1 - \frac{nonzero\ entries}{total\ entries}$.

The main purpose in this study as mentioned earlier is to show that association mining can be used to analyze service order data and the rules found by association mining can be utilized for recommender systems in predicting customer preferences. Two different association mining based recommender systems are used in this paper. They are briefly described in the following section together with the baseline random and naïve methods.

3 Recommender Systems

Market basket analysis can provide tremendous benefits for retailers and other commercial entities. But the main assumption in market basket analysis is that all the items are purchased together in the same transaction. Association mining results are generated usually based on this assumption. This assumption may not be valid to explain the whole picture in the service organizations. Because transactions also involve with current products and services in the customer profiles.

A simple recommender system based on association mining can be built by matching rules with the customer profiles. Usually product(s) on the right hand side of the rules are recommended to those customers whose profiles satisfy the left hand side of the rules. In this case, an exact match is required.

An extension to the exact match method is to use a similarity measure between customer profile and the rules. Both methods are discussed in an unpublished paper by the author [1]. Thus exact match and similarity based recommender systems are used in this paper to utilize association rules. By the nature of the problem defined in Section 2, association rules need to be screened before used by recommender systems. In this step, association rules are screened to have only one added product on the right hand side and retained, deleted, changed products on the left hand side.

To assess the predictive ability of association mining based recommender systems, random and naïve methods are also used in this work as baseline methods. In both approach, recommendations are made based on frequency of the products in the training data. In random recommendation, five products are sampled without replacement based on the distribution of their frequencies. We also report results from random recommendations made based on most frequent twenty products. Naïve method recommends five most frequent available products for a given customer profile.

Using sparse data format in product recommender systems improves the scalability significantly as reported in [2]. All the five methods mentioned above are implemented in a sparse data format fashion. We report the results from these five different methods in the following section.

4 Experimental Results

In this section we report the performances of five different recommender systems. A successful prediction is defined as recommending an added product within the five recommendations. All methods make recommendations based on the same training data. Test data is provided to all of the methods without added products. A successful transaction is defined as predicting at least one added product correctly for that transaction. Each recommender system is allowed to make up to the five recommendations. The results are summarized in Table 2 and Figure 1. Accuracy rate represents the ratio between the number of successful predictions and the number of added products. Success rate is defined as the ratio between successful transactions and number of transactions. We plotted the log of number of successful predictions at each prediction level in Figure 1.

As seen from the results that association mining based recommender systems performed significantly better. Note also that as Table 1 indicates, the training and test data might have totally different frequency distributions. In the training data, five most frequent products correspond to 31.72% of the total additions. On the other hand, five most frequent products correspond to 78.85% of the total additions in the test data. Frequency based methods missed the many transactions. This is a very valid scenario for the live recommender systems. Historical distributions may not reflect the short time fluctuations. In a sense, frequency based methods are overtrained and can not generalize well due to the high bias towards historical data.

Moreover, association mining results might cause poor performances due to the less coverage of product base. Meaning that we may not have association rules for many of the added products. For this particular data, there were 199 distinct added products in training data but association mining generated rules for 18 of them at the 40% confidence and 0.4% support levels. Although there were 140 distinct added products in the test data, both the exact match and the similarity methods made recommendations for all of the 18 products.

According to empirical results given in [1], it is better to have a confidence and support level as low as possible. But the accuracy of the predictions mainly depends on the confidence level. As we mentioned above a 40% confidence and 0.4% support level are used in this work.

We plotted the log of the number of successful predictions at each prediction level in Figure 1. Association mining based recommender systems were able to predict a significant portion of the added products in the first two recommendations. Random recommender systems showed a uniform behavior. Since the sequences of the frequent added products are different in both training and test data, naïve recommender system had a hike in the fourth recommendation.

5 Related Work

Indirect [5] and negative [4] association mining studied in the literature before. In indirect association, two items might have strong associations with a third one individually, but they may not have a direct associations between them [5]. If we use association mining on purchase transactions, we can end up with such indirect associations among substitute products.

In the case of negative association [4], assume that there two substitute products. A third item might have a strong association with one of them but not with the other one. This constitutes a negative association.

Analyzing stock market trends using association mining is also similar to the discounted products example given above. A positive change in one particular stock might cause a negative change on another one. Similarly it might also cause a positive change on another stock. But these associations among stocks can also correspond to the dependencies among them. This point should be addressed properly.

Association mining is also used for classification purposes in the literature [3]. In our context, states can be considered as different classes of items. We can also loosely incorporate this idea. But it should be investigated carefully. More thoughts are needed in this direction.

Table 2: Result from Recommender Systems

Statistics	Recommender System				
	Exact Match	Similarity	Random	Random-20	Naïve
Successful Prediction	33342	33755	5638	8376	12701
Accuracy (%)	79.31	80.30	13.41	19.92	30.21
Successful Transactions	22964	23342	5363	7887	12089
Success Rate (%)	82.69	84.05	19.31	28.40	43.53

6 Conclusion

We successfully analyzed the service order data using association mining. Besides predicting the added products (product recommendation), current system can also be utilized to predict the final state of a customer profile after a service order transaction. This can be done easily by simulating the each item in a customer profile for different states. The rule with the highest confidence level can provide information about the final state of the customer profile after a possible service order transaction.

Item-based product recommender systems usually generate excellent prediction performances [1, 2]. It would be interesting to investigate the performances of those recommender systems using totally different training and test data.

Acknowledgments

This study was conducted under CRM efforts in Verizon by EDesk Team. I would like to thank to every single member of EDesk team, especially Moid Khan creating the data gathering tool.

References

[1] A. Demiriz. Enhancing product recommender systems on sparse binary data. Under journal review, February, 2002.

[2] G. Karypis. Evaluation of item-based top-n recommendation algorithms. In *Proceedings of the Tenth International Conference on Information and Knowledge Management (CIKM)*, Atlanta, GA, 2001.

[3] B. Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining. In *Proceedings of Fourth International Conference on Knowledge Discovery and Data Mining*, pages 80–86, New York, NY, 1998.

[4] A. Savasere, E. Omiecinski, and S.B. Navathe. Mining for strong negative associations in a large database of customer transactions. In *Proceedings of the 14th International Conference on Data Engineering*, pages 494–502, 1998.

[5] P-N. Tan, V. Kumar, and J. Srivastava. Indirect association: Mining higher order dependencies in data. In *Principles of Data Mining and Knowledge Discovery*, pages 632–637, 2000.

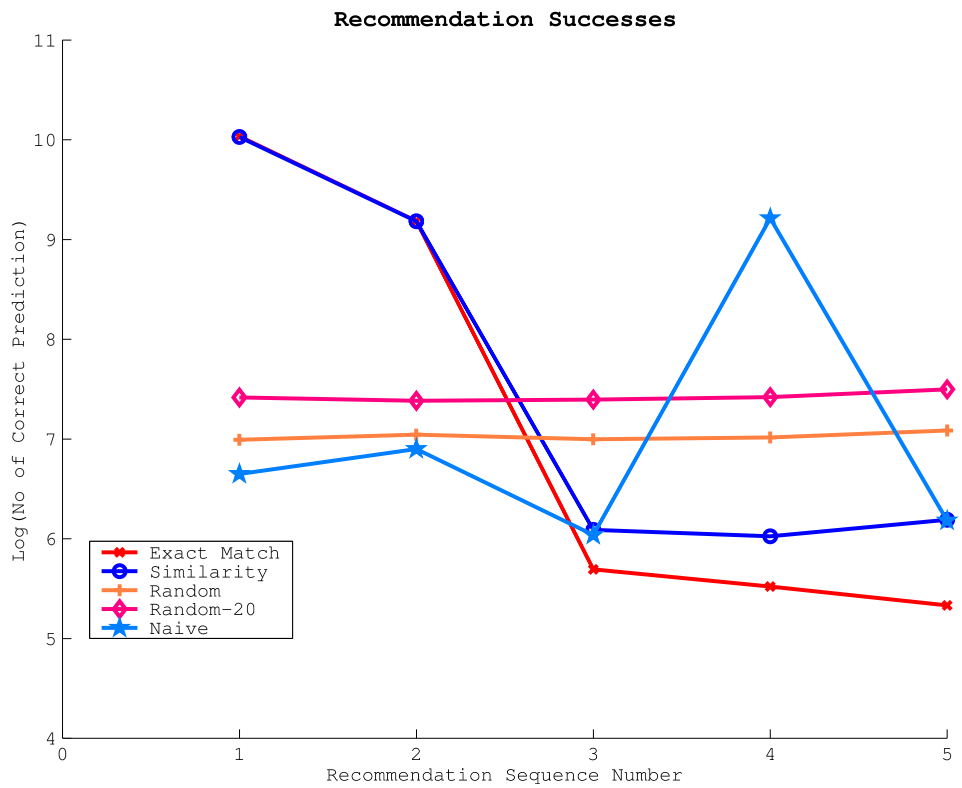

Figure 1: Comparison of Recommender Systems