Finding Similar Time Series in Sales Transaction Data

Swee Chuan Tan(✉), Pei San Lau, and XiaoWei Yu

School of Business, SIM University, 535A Clementi Road, Singapore, Singapore
jamestansc@unisim.edu.sg, {jessenie,iewoaixuy}@gmail.com

Abstract. This paper studies the problem of finding similar time series of product sales in transactional data. We argue that finding such similar time series can lead to discovery of interesting and actionable business information such as previously unknown complementary products or substitutes, and hidden supply chain information. However, finding all possible pairs of n time series exhaustively results in $O(n^2)$ time complexity. To address this issue, we propose using *k-means* clustering method to create small clusters of similar time series, and those clusters with very small intra-cluster variability are used to find similar time series. Finally, we demonstrate the utility of our approach to derive interesting results from real-life data.

1 Introduction

The term market basket analysis in data mining [2, 5] involves analyzing and understanding products and their categories that are commonly purchased by a customer over a specific period of time, e.g., during a visit to a departmental store. This is an intriguing research topic because the insight derived from this kind of study can be very useful.

In the retailing context, identifying complementary products (e.g., coffee and sugar) or substitute products (e.g., sugar and artificial sweetener) can help a marketer to design better promotion campaigns to increase sales of related product lines. Sales managers can also use such information to formulate better pricing strategies that could lead to improved market share or profits. Supermarket store managers can use insight from a market basket analysis to improve store layout and design, so that commonly purchased items can be more easily accessed by shoppers.

Obviously, the notion of 'market basket' can also be extended into many other areas beyond the retailing space. For example, a pharmacy manager can use the information to organize storage of medicines so as to improve the efficiency of locating, retrieving, and even packaging of some medicinal drugs commonly used to treat a certain disease.

Despite all the potential benefits we can get from market basket analysis, the analytics task per se is not straight forward and a lot of research is still ongoing. In marketing science, the common approach is to use traditional statistical techniques to analyze customer purchases. Most of such studies aim to improve our general understanding of customer purchasing behaviors (e.g., see [8, 17]).

M. Ali et al. (Eds.): IEA/AIE 2015, LNAI 9101, pp. 645–654, 2015.
DOI: 10.1007/978-3-319-19066-2_62

In the area of data mining applications, the problem is more applied in nature because the analyst is usually asked to solve a particular business problem at hand. For example, given a sales transaction dataset, the problem is usually to draw insight from commonly purchased products so as to improve sales and operations in the near future. To this end, the association rule mining (ARM) approach proposed [1] two decades ago has been a very important topic frequently studied by data mining researchers.

In this paper, we are motivated to tackle the 'market basket analysis' problem using a time series clustering approach instead of the ARM approach. This is because the proposed approach uses time series of product sales as records of interest, in which quantity and time information are used in the analysis, yielding richer and more insightful results. Furthermore, time series patterns are generally more concise as compared to large number of rules produced by ARM. Indeed, previous research has shown that storing and analyzing data in time series format actually takes up less computer memory space, compared to the amount of memory space used to store binary format data in ARM [16].

Another advantage of finding similar time series of product sales is that, it is easy to study how each time series sales pattern is associated with a certain entity of interest (e.g., with a particular customer). The entity then provides a unique context in which potentially useful domain information can be used to interpret the time series patterns so as to produce meaningful insight. Here we show two example entities that could be associated with time series patterns.

The first type of entity is *product category*. We can study the time series of product sales patterns associated with product categories. If two products are in the *same product category* (e.g., tooth paste) and are having similar time series patterns of product sales, then they could potentially be *substitute* products. On the other hand, if two products are in *different product categories* (e.g., tooth paste and tooth brush) and are having similar time series patterns of product sales, then they could potentially be *complementary* products.

The second type of entity is *customer making purchases*. We can study the time series pattern of the sales of a product purchased by a particular customer. If there are two similar time series patterns of two different customers purchasing the same product, then the pattern suggests that these two customers may have similar demand on the same product. This implies that these two customers could be related, such as working on a common project.

The rest of this paper is organized as follow. Section 2 reviews related work. Section 3 presents the general proposed approach to find similar time series in sales transaction data. Section 4 illustrates how the approach is used for analyzing a sales transaction dataset containing more than 300 thousand records. We then present a case where real-life insights were discovered using our approach. We also show how similar time series can be discovered from the dataset while association rule mining fails to produce useful results. Finally, Section 5 concludes this paper by discussing some future research directions.

2 Related Work

The first association rule mining technique that has been proposed in the literature, known as the *Apriori* algorithm [1], is also a tool commonly used by data miners. Despite its popularity, Apriori has several weaknesses that are still not well addressed today.

The first weakness arises from the fact that the *Apriori* algorithm requires the purchase status of every product associated with a transaction to be recorded in binary format, e.g., Product_A_Purchased = {Yes, No}. Hence the mining problem is sometimes referred as the Boolean Association Rules problem [14]. This form of data representation gives rise to several issues. Firstly, the analysis cannot take quantity information into account. In fact, quantitative variables have to be discretized either during the preprocessing stage or during the rule mining process [14]. Apart from losing information, the need to use discretization will increase the complexity of the rule mining process because the user has to choose an appropriate discretization method and find the right parameter settings (e.g., the number of categories). Despite many attempts (e.g., [9, 11, 13, 18]), there is still no established approach to address this problem. Secondly, Boolean Association Rules problem does not consider the time dimension [12]. As such, the method cannot discover rules that may only be interesting in certain time window (e.g., Christmas period) but uninteresting in other times. To address this problem, some authors propose solving the temporal data mining problem [3], in which the main difficulty lies in finding the time window that contains interesting rules.

The second weakness of Apriori is that it tends to generate a huge number of rules. It is not uncommon to see Apriori (and in fact many other ARM methods) producing hundreds or thousands of rules from a dataset with only a few hundred transactions. This poses a serious problem to the data miner because there is a risk that some 'interesting' rules could have happened just by chance alone [19]. To the user, it is also difficult to find the few most interesting rules among many uninteresting ones. Currently researchers are looking at possible solutions such as rule reduction (e.g., [7]), concise rule generation (e.g., [10, 20]), rule visualizations (e.g., [4]), etc. Such research directions help move towards developing a more usable rule mining system, but as of now many business users are still facing difficulty in using existing systems.

In this paper, we explore an alternative approach that is easier to implement and use compared to association rule mining. This alternative market basket analysis approach involves time series clustering on real-world sales transaction data. We show that this approach can help alleviate some of the above-mentioned issues.

3 Proposed Approach to Find Similar Time Series

Here we describe the steps to prepare data for time series clustering and the procedure to find similar time series.

Given a dataset that contains a finite number of transaction records over a specified time period T. The time domain T can be represented as v time units, $T = \{0, 1, 2, \ldots, v-1\}$. When a transaction occurs at time i, the transaction records the quantity (q_i) of a product (p_i) being purchased at time i. Let $T = \{w_0, w_1, \ldots, w_{m-1}\}$ be made up of m

non-overlapping time windows. Then the j^{th} time window w_j having d consecutive time units will have time units $w_j = \{j \times d, j \times d + 1, \ldots, j \times d + d - 1\}$. Once d, the window width is defined, we also know that $m = v/d$, and the total quantity of a product P ordered in time window w_j is $Q_j = \sum q_i \times \mathrm{I}(p_i \in \mathrm{P})$, for all i that satisfy the constraint $j = \lfloor i/d \rfloor$. Note that $\mathrm{I}(p_i \in \mathrm{P}) = 1$ if $p_i \in \mathrm{P}$; otherwise $\mathrm{I}(p_i \in \mathrm{P}) = 0$.

Eventually, we can represent the time series of a product P as a tuple $S := \{P, Q_0, Q_1, \ldots, Q_{m-1}\}$. Note that the quantities ordered for different products can be quite different, so each total ordered quantity value in a time window should be normalized to a range of [0, 1]. This is achieved using min-max normalization, $Q^*_j := (Q_j - Q_{min})/(Q_{max} - Q_{min})$, where $Q_{min} := min(Q_0, Q_1, \ldots, Q_{m-1})$, and $Q_{max} := max(Q_0, Q_1, \ldots, Q_{m-1})$. This gives an equal weight to all product sales time series and facilitates clustering of similar time series structures. Note that S is excluded if $Q_{max} = Q_{min}$.

Once the time series data is prepared, we use a four-step procedure to find similar time series.

- Step 1: Given x time series, decide the expected number of time series (y) to be included in each cluster.
- Step 2: Apply *k-means* using $k = \lfloor x/y \rfloor$, where k is the number of clusters.
- Step 3: For each cluster, sort each time series by its distance to the centroid.
- Step 4: Select all consecutive pairs of ordered time series in each cluster that has similarity greater than some predefined threshold.

Note that our main aim is very different from the most common intent of cluster analysis – to find most natural groupings. Instead, our aim here is to find many (small) groups of similar time series. Hence the choice of y is not critical in Step 1, and we recommend it to be a small number, say 5. Once y is determined, k can be computed and *k-means* [6] can be applied. In Steps 3 and 4, we use the centroid regions to find the most similar time series as these are areas where highly similar time series reside.

4 Finding Similar Time Series in Sales Transaction Data

We demonstrate the utility of the proposed approach using two cases. The first case involves analyzing a real-life sales transaction dataset that we have purchased for academic research purposes. We present selected time series clustering results derived from this dataset to demonstrate how the proposed method works. We shall call this dataset "Sales Transaction".

The second case involves analyzing a sales transaction dataset obtained from a global part supplier. The actual identity of this company cannot be disclosed due to confidentiality reasons. We shall call this company as "Company Z". Although we are not allowed to show the actual time series patterns of Company Z data, we are allowed to reveal some insights derived from the analysis.

Data Preparation of the Sales Transaction data: The Sales Transaction dataset contains 350 thousand transactional records of about 800 products purchased over a

period of one year. For the purpose of time series clustering, we only require the Product Code, Sales Date, and Quantity Ordered of each transaction.

During the data preparation stage, the transaction records are sorted by Product Code and then Sales Date. The records are then aggregated by a two-week time window and the total quantity ordered for each product within each two-week period is computed. The result of aggregation is a table with Product Code and Window Period Identifier (an integer than ranges from 1 to 26), and Period Sales Quantity. This table is then transposed to have each Window Period Identifier representing a column capturing the total sales quantity of each product for the period. The resulting table ends up with about eight hundred records over a period of 26 time-points. Each record captures periodic sales quantities of a product sold over the one-year period.

Figure 1 gives an example of the data conversion from sales transaction format to time series format. Since the data spans over only 26 time points, the data size is small and we can apply clustering directly on the time series data.

Once the time series data is prepared, the sales quantities for each product are then normalized to a range of [0, 1] using the above-mentioned min-max normalisation. Finally, the dataset is imported into the IBM SPSS Modeler®, which is the software package used for this study.

Product Code	Period ID	Sales Qty
D23	1	73
D23	2	59
D23	...	...
D23	26	67
D24	1	15
D24	2	22
D24	...	...
D24	26	23
...	...	...

	Period ID			
Product Code	1	2	...	26
D23	73	59	...	67
D24	15	22	...	23
...	...	...	...	...

Fig. 1. Data conversion from sales transaction table format to time series format

Analysis of the Sales Transaction Data: In this project, there are about 800 products in the dataset; the expected number of time series in each cluster is set as 4. As a result, we use *k-means* clustering algorithm to generate 200 clusters. The following presents how interesting business information can be derived from the results.

Since most of the patterns found are similar, we present some typical example patterns to highlight the key insight. The first example pattern is presented in Figure 2, which shows a positive relationship between Product 281 (Product Category: HiFi) and Product 293 (Product Category: Telephony) found in Cluster 117. Notice that these two products are from different product categories. This result suggests the first possible insight that may be derived from this type of time series pattern: *Two products in different product categories could be complementary products if they have similar product sales time series.*

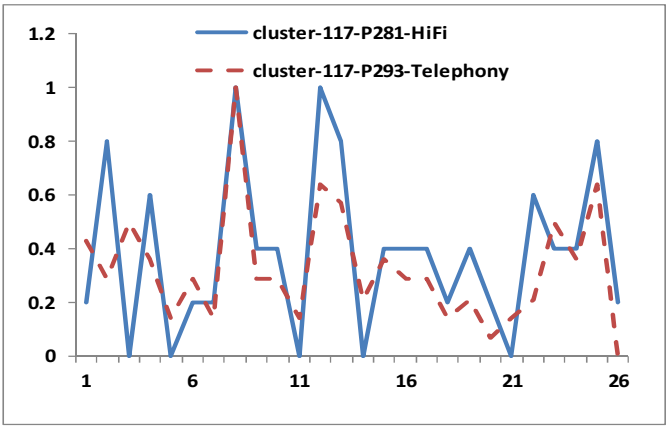

Fig. 2. Similar time series of products sales in different categories suggests opportunities for cross-selling

The second example pattern is presented in Figure 3, which shows a positive relationship between Products P408 and P613. Upon further investigation, it is found that these products all belong to the Hardware product category. In huge procurement systems, the actual relationships amoung thousands of products are usually hidden. Sometimes, previously unknown substitute products could be discovered this way. This result suggests the second possible insight that may be derived from this pattern: *Two products in the same product category could be substitutes products if they have similar product sales time series.*

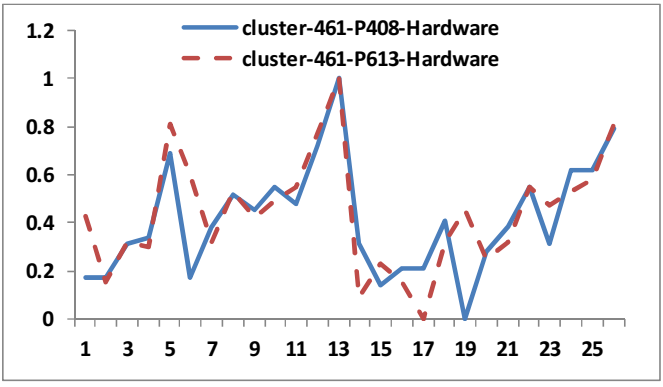

Fig. 3. Similar time series of products sales in the same category suggests opportunities for understanding the reason(s) for the demand

Sometimes, time series may be similar due to coincidence. It is important to validate every possible insight with the domain expert. The above examples also suggest that the product sales time series should be interpreted within a predefined context. In these examples, knowing whether two products are in the same or different product

categories suggests that the products are substitutes or complementary, respectively. In the following, we present a case study where similar product time series, when interpreted in different contexts, result in different types of insights that are useful for decision making.

Case Study of Z Company: Here, we report a case study that has been conducted with a real company. Due to confidentiality reasons, we cannot reveal the name of the company, and shall call it **Company Z**. This company is a supplier of raw parts to many manufacturing companies around the world. At the time of the study, the company wanted to better understand their customers' needs in order to better manage their productions, stock control and customers. One of the initiatives towards this end was to analyze the product sales transaction data in order to discover interesting customer purchasing behaviors. In fact, some early results of Company Z have been presented in previous papers [15, 16]. Here we present more recent results derived from the same project.

The first type of discovery involves grouping the product sales time series according to products ordered by the *same customer*. In doing so, we examined a few hundreds of raw parts (e.g., nuts and screws) that have been ordered by buyers of *a single manufacturing plant*. Using the proposed method, we were able to find some groups of parts that exhibit similar product sales time series. Subsequently, the customer confirmed that a number of these parts with similar time series are indeed complementary parts for manufacturing certain consumer electronics products. Armed with such knowledge, we were then able to advise the sales team on pricing and sales strategies of these parts.

The second type of discovery involves grouping the product sales time series according to products ordered by *different customers*. Sometimes, this type of analysis can reveal previously unknown interactions among customers. In this case, we found that certain parts purchased by two different customers, namely Company B and Company C, did have very similar time series patterns. On further investigation, it was found that Companies B and C were actually supplying products to another customer, Company A. It is interesting to note that Companies A, B and C are all customers of Company Z (i.e., the raw part supplier). The supply chain involving companies A, B and C are illustrated in Figure 4.

Figure 4 shows that Company Z has been supplying a variety of products to three companies. Company A is a special product maker for automotive market. Companies B and C are competitors – both provide value-added services for raw materials used in white goods. This finding is counterintuitive because Company A is known to serve a market that is different from the market served by Companies B and C.

To ensure the validity of this finding, we talk to Company A and confirm that Companies B and C are indeed their qualified vendors for parts used in making certain products. Companies B and C are chosen due to their expertise in upgrading parts to be used in their products, and because of their factory locations, they are able to offer shorter lead-time for goods delivery.

This finding suggests that the engagement of these companies can now be more organised. Currently, Companies A, B and C are handled by three different sales staff

as these companies are perceived as serving different markets. However, the clustering result reveals that these companies all serve the same market within the discovered supply chain. This is valuable information for sales and operations. For example, the company could now assign one sales staff (instead of three) to take care of all three companies. This is likely to result in a more productive and consistent engagement of these companies. Other issues such as (fairness in) pricing, sharing of sensitive information, and identifiifcation of cross-selling opportunities can now be beter managaed by one (and not three) sales staff.

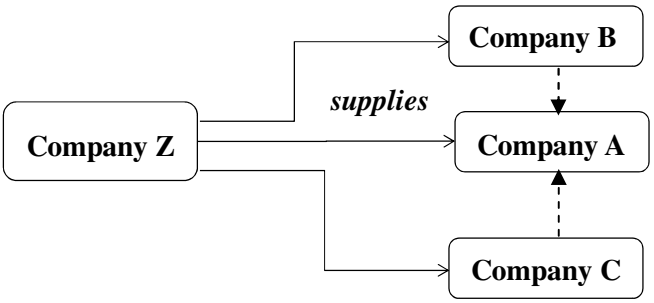

Fig. 4. Hidden supply chain information (the dotted lines) discovered from our analysis. Companies B and C are competitors supplying to Company A. All Companies A, B and C are customers of Company Z.

Applying Apriori on the Data: We apply the *Apriori* method on the Sales Transaction dataset using a minimum antecedent support threshold of 5% and minimum confidence threshold of 20%. This produces 55 rules with very low support and confidence values. Figure 5 shows a scatter plot of support and confidence percentage of these 55 rules. It is easy to see that the all rules have confidence lower than 25% and support lower than 4%. This suggests that association rule mining is unable to discover any interesting rules from the data.

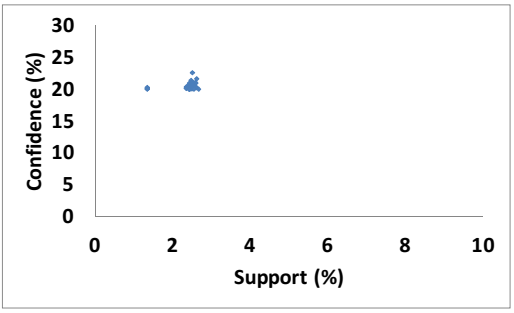

Fig. 5. The 55 rules generated from the Sales Transaction data set are all of low support and low confidence

5 Concluding Remarks

This paper shows that finding similar time series of sales data can help reveal interesting information about customers. We have shown that it is possible to discover complementary or substitute products that can lead to better formulation of sales strategies. Also, it is possible to discover hidden processes that are important information for implementing better customer engagement. Our discovery of hidden supply chain information is a case in point.

One question that arises from this study is that why the *Apriori* method is unable to produce any interesting rules from the given sales transaction data? One reason is because the combinations of individual products are not purchased often enough by individual customers. However, when we examine the time series pattern of the total sales quantity of each product over different time windows, we do find relationship between certain products that has similar time series patterns. One caveat against misusing such results is that we should always check that the purported relationship of any two products is not due to coincidence, and this can be easily verified with the domain expert.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proceedings of the International Conference on Very Large Data Bases, pp. 487–499 (1994)
2. Charlet, L., Kumar, A.: Market Basket Analysis for a Supermarket based on Frequent Itemset Mining. International Journal of Computer Science Issues. **9**(5), 257–264 (2012)
3. Chen, X., Petrounias, I.: Discovering Temporal Association Rules: Algorithms, Language and System. In: Proceedings of the 16th International Conference on Data Engineering, pp. 306–306 (2000)
4. Hahsler, M., Chelluboina, S.: Visualizing association rules in hierarchical groups. Presented at the 42nd Symposium on the Interface: Statistical, Machine Learning, and Visualization Algorithms (Interface 2011). The Interface Foundation of North America, June 2011 (unpublished)
5. Kim, H.K., Kim, J.K., Chen, Q.Y.: A Product Network Analysis for Extending the Market Basket Analysis. Expert Systems with Applications **39**(8), 7403–7410 (2012)
6. MacQueen, J.B.: Some Methods for classification and Analysis of Multivariate Observations. In: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability 1, pp. 281–297. University of California Press (1967)
7. Mafruz, Z.A., David, T., Kate, S.: Redundant association rules reduction techniques. International Journal Business Intelligent Data Mining **2**(1), 29–63 (2007)
8. Manchanda, P., Ansari, A.: Gupta, S: The ``shopping basket'': A model for multicategory purchase incidence decisions. Marketing Science **18**(2), 95–114 (1999)
9. Minaei-Bidgoli, B., Barmaki, R., Nasiri, M.: Mining numerical association rules via multiobjective genetic algorithms. Information Sciences **233**, 15–24 (2013)
10. Palshikar, G., Kale, M., Apte, M.: Association Rules Mining Using Heavy Itemsets. Data & Knowledge Engineering **61**(1), 93–113 (2007)

11. Tong, Q., Yan, B., Zhou, Y.: Mining Quantitative Association Rules on Overlapped Intervals. In: Li, X., Wang, S., Dong, Z.Y. (eds.) ADMA 2005. LNCS (LNAI), vol. 3584, pp. 43–50. Springer, Heidelberg (2005)
12. Qin, L.X., Shi, Z.Z.: Efficiently Mining Association Rules from Time Series. International Journal of Information Technology. **12**(4), 30–38 (2006)
13. Salleb-Aouissi, A., Vrain, C., Nortet, C., Kong, X., Rathod, V., Cassard, D.: QuantMiner for Mining Quantitative Association Rules. Journal of Machine Learning Research **14**, 3153–3157 (2013)
14. Srikant, R., Agrawal, R.: Mining Quantitative Association Rules in Large Relational Tables. In: Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pp. 1–12 (1996)
15. Lau, P.S.: Time Series Clustering for Market Basket Analysis. Analytics Project Report. SIM University (2013)
16. Tan, S.C, Lau, P.S.: Time Series Clustering: A Superior Alternative for Market Basket Analysis. In: Herawan, T., Deris, M.M., Abawajy, J. (eds.) Proceedings of the First International Conference on Advanced Data and Information Engineering. LNEE(LNCS), vol. 285, pp. 241–248. Springer, Singapore (2013)
17. Vindevogel, B., Poel, D., Wets, G.: Why promotion strategies based on market basket analysis do not work. Expert Systems with Applications **28**(3), 583–590 (2005)
18. Webb, G.: Discovering associations with numeric variables. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2001), pp. 383–388. ACM, New York (2001)
19. Webb, G.: Discovering Significant Patterns. Machine Learning **68**(1), 1–33 (2007)
20. Xu, Y., Li, Y., Shaw, G.: Reliable representations for association rules. Data & Knowledge Engineering **70**(6), 555–575 (2011)