2018 IEEE International Conference on Data Mining Workshops (ICDMW)

Effective Steering of Customer Journey via Order-Aware Recommendation

Joël Goossens
Analytics for Information Systems Group
Eindhoven University of Technology
Eindhoven, The Netherlands
Email: j.a.j.goossens@student.tue.nl

Tiblets Demewez
Data Science Department
Philips Research
Eindhoven, The Netherlands
Email: tiblets.demewez_1@philips.com

Marwan Hassani
Analytics for Information Systems Group
Eindhoven University of Technology
Eindhoven, The Netherlands
Email: m.hassani@tue.nl

***Abstract*—The analysis of customer journeys is a subject undergoing an intense study recently . The increase in understanding of customer behaviour serves as an important source of success to many organizations. Current research is however mostly focussed on visualizing these customer journeys to allow them to be more interpretable by humans. A deeper use of customer journey information in prediction and recommendation processes has not been achieved. This paper aims to take a step forward into that direction by introducing the Order-Aware Recommendation Approach (OARA). The main scientific contributions showcased by this approach are (i) increasing performance on prediction and recommendation tasks by taking into account the explicit order of actions in the customer journey, (ii) showing how a visualization of a customer journey can play an important role during predictions and recommendations, and (iii) introducing a way of maximizing recommendations for any tailor-made Key Performance Indicator (KPI) instead of the accuracy-based metrics traditionally used for this task. An extensive experimental evaluation study highlights the potential of OARA against state-of-the-art approaches using a real dataset representing a customer journey of upgrading with multiple products[1].**

***Index Terms*—Business intelligence, Process mining, Recommender systems, Behavior Mining, Customer Journey.**

I. Introduction

In today's society, the interactions a customer has with an organization are quite plentiful thanks to the myriad of ways in which these customers are now able to interact with the organizations. These interactions can be seen as a sequence of events, where in each, the customer achieves a certain goal with a specific interaction. Such a sequence of observed events belonging to a single customer is referred to here as a customer journey. The analysis of such customer journeys can be a huge boon towards improving the organizations, as the key objective is to get an understanding of how the experiences of the customer can be enriched though what marketers call the decision-making process [1].

To properly interpret the customer journey data that organizations possess, it is helpful to create a visualization of this information to get an idea of which steps are usually taken in the journeys. Such a representation is called a customer journey map. These artefacts often possess a non-linear structure while reflecting behavioural, emotional and cognitive drives [2]. A mapping in this paper is obtained by means of process mining [3]. The result is known as a process model, of which an example is shown in Figure 1. The example shown here is from the website of a music festival. Firstly, a customer will have to register themselves. Upon completing the registration, they go on to either buy tickets or merchandise from a band. In case tickets are bought, it might occur that the customer wishes to also buy tickets from another band, which is indicated by the arrow to and from itself. A customer is able to end their journey after taking either of these actions, but it might also be the case that they still need to change part of their information, for example their payment credentials. This information can then be altered and afterwards everything is in order to deliver the tickets and/or merchandise, leading to the customer journey end. Note that in this example customers are only doing a single purchase, but it might also be the case that one wishes to model all purchases made by a single customer in which case the process model would be different and more complex.

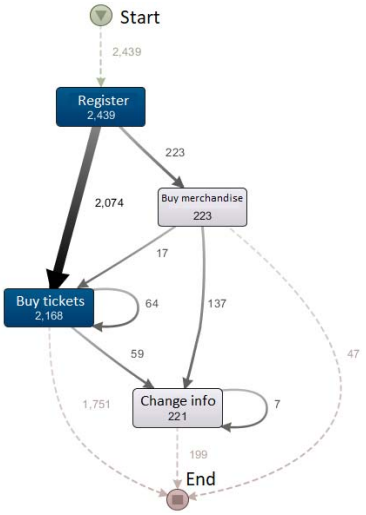

Fig. 1: Example process model of a customer journey.

The approach proposed in this paper is called the Order-Aware Recommendation Approach, shortened to OARA, and

[1]Most of the second author's contribution to this work was done as she was with Signify™(previously Philips Lighting).

2375-9259/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDMW.2018.00123

aims to improve upon the current state of the art in three areas. Firstly, the extraction of a customer journey map by means of process mining is a technique which hs been recently contributed in [3], [4]. However, the approach proposed here aims to go beyond simply extracting a model. The extracted model is now used by OARA to also perform predictions and recommendations for future steps in the customer journeys in a tailor-made manner. This allows for the value of customer journey data to rise as a result, as one can now rely on machine learning techniques for these tasks which would otherwise involve a large amount of manual labour. Furthermore predictions and recommendations on the customer journeys can also be done by other predictor algorithms and recommendation systems since one still would like to obtain information on the future based on the past. OARA however aims to improve upon the existing methods as they do not take the explicit order information into account which is present in the customer journeys. This loss of contextual information can then lead to a decrease in quality compared to when this context is applied to the predictions and recommendations. Additionally, we address the issue of evaluating recommender systems which mainly focuses in most of the available work on prediction accuracy. Other evaluation properties such as novelty are less explored [5]. This is a mismatch with reality as the goal organizations usually have when recommending an action to customers is the maximization of a Key Performance Indicator (KPI). This is a value which measures how well an organization is performing on a specific key objective. To provide a solution to the current situation it is shown how one can take KPI maximization into account by using OARA.

To summarize, the contributions made by this paper are investigating the possibilities to:

- Use a process model to do predictions and recommendations within the customer journey.
- Explicitly use the order of events during predictions and recommendations.
- Optimize recommendations for any chosen KPI.

The rest of this paper consists of the following sections: Firstly some related work for the rest of the paper is presented in Section II. In Section III, the context and problems tackled in the paper are concretely defined, in Section IV, it is explained in detail how OARA allows for the recommendations to be created. In Section V, a real dataset is used for an evaluation of the quality of both the predictions and recommendations provided by OARA and to wrap everything in Section VI the conclusions and future research opportunities are given.

II. Related Work

In this section some related work from different areas is introduced. Firstly some more context is given regarding the concept of customer journeys, after which process mining is discussed. Following upon this stream data mining is covered and lastly the Recency, frequency and monetary value is introduced, which is later used for segregating the customer base.

A. Customer journeys

As described in [6], the term customer journey is quite widely used in scientific literature with yet no common understanding exists with regard to what a customer journey exactly entails. Descriptions used in the past include that a customer journey is the cumulation of repeated interactions between the service provider and customer [7], an ”engaging story” based on the interactions of a customer with the service [8], or a ”walk in the customer’s shoes” [9]. All descriptions have in common that a high importance is placed on the experience of the customer. The approach proposed in this paper aims at using the logged events to recreate this experienced journey and using the distinguishing qualities which lie inside them to achieve high quality predictions of customers future interactions and recommendations regarding the best possible interaction from both user and organization perspectives.

A combination of research between the research fields of process mining and customer journeys has occurred in the past [4], where the goals was to extract customer journey mappings. These are a visualization of the customer journey, and in that research the events which were relevant to the customer journey scenario were retained such that a process model could be created upon them. This process model was used both for further analysis tailored to process mining as well the creation of the customer journey mapping. The research in this paper also uses such a mapping towards a customer journey while taking care of two tasks in the proposed future research of this paper: finding techniques for clustering customer journeys and facilitating predictions on future behaviour in these journeys. Recently, [3] proposed an idea to optimize the customer journey recommendations by using process mining. In this work, we are presenting an end-to-end framework by introducing a different order-aware recommender, introducing a new evaluation measure for recommendations, and performing en extensive experimental evaluation on a real-life dataset.

B. Process mining

Process mining is a research area which combines the domain of process modelling and analysis with the domain of data mining and machine learning. The goal of this combination is to discover, monitor and improve processes based on knowledge from data which is stored in the event log format regarding the process in question [10]. Event logs show the occurrence of events at a designated point in time, where the event is an action logged by an information system such as the sale of a product. This event is specified to have come from a specific process or instance, also known as case [11]. One such instance or case then encompasses all events belonging to a single customer which can be identified based on an ID.

Process mining mainly plays a role in helping OARA to determine which information and activities should be included during the predictions and recommendations on the customer journeys. Ideally, one is able to find an easy to understand model which shows the process from a high level, as exemplified in Figure 1. Here the overall process is short

and intuitive, but there are also cases where the number of events per instance is very large while there are also connections between almost all of such events. In this case the model becomes entangled and as a result hard to interpret which can be counteracted by taking the most representative samples and segregating the customer base. In Figure 1, using process mining notation will result in the following: a single event in the customer journey called for example $Register$ has the following combination of information: $Register = (c, a, t)$. c here stands for the case, which is a specific customer, a is the action performed, registering, and t the time at which the action was performed. A customer journey consists of multiple such events and is then denoted as $CJ = \langle Register, BuyTickets \rangle$, where $Register$ and $BuyTickets$ are events belonging to the same case ID c have consecutive timestamps. The entire collection of journeys is here equivalent to an event log and is denoted as $Log = \langle Register, BuyTickets \rangle, \langle Register, BuyMerchandise, ChangeInfo \rangle, \langle Register, BuyTickets, BuyTickets, ChangeInfo \rangle$. Note that based on the presence of a loop there is no exhaustive Log which covers all possible customer journeys and that journeys belonging to different customers might be interleaving depending on t.

C. Stream data mining

The environment described in this paper is one where the information is obtained by means of data streams, which can be characterized as continuous and typically non-constant [12]. There are two main issues which arise from such data streams. Firstly these streams produce massive, potentially infinite, amounts of data which can make it hard to use more time consuming operations on such data. Secondly the information in the data can change rapidly, which makes it important to facilitate an option for fast updates. OARA aims to cover these issues by taking a collection of 'base' information on which it builds while having parts which can be updated with new information. Several recommenders have been proposed which are able to adapt to these circumstances. One of them is called OCuLaR [13]. The aim of of this approach is to generate recommendations which are easily interpretable by the customers based on data where there is implicit feedback, i.e. no information is supplied by the customers regarding their enjoyment on or motivations for choosing certain products. One thing to note about this approach is that it does not use any features, it only considers relationships based on the products customers bought, and as such does not explicitly use any context information during the recommendation. The approach proposed in this paper will on the other hand do this based on the hypothesis that there lies important knowledge in the context which can be used to amplify the predictive qualities.

D. Recency Frequency Monetary Value

The Recency, Frequency and Monetary values, often shortened to RFM, is a KPI based on how well a customer performs in the recency, frequency and monetary dimensions which has been introduced in [14]. Recency here means the time interval which has passed between the previously observed interaction of the customer and the present, Frequency involves how often a customer has interacted with the organization, possibly within a specified time period, and Monetary value is based on the cumulated amount of money the customer has spent at the organization. The RFM values are used in the case study of this paper to segregate the customer base, which was inspired by earlier successes such as reported in [15] and [16]. The notation for the groups here will be as follows: *XYZ*, where *X* signifies if the Recency was relatively high or low, and the same is signified by *Y* with regard to the Frequency and *Z* for the Monetary value. As such, for example one can have the *HLH* group where an event was recently observed, relatively few events were observed in total and the monetary value of the steps taken by the customer is relatively high due to the few purchases which were observed involving more expensive products.

III. Problem description and Setting

Firstly, the type of customer journey is described here to get an idea for what kind of environments the approach explained in this paper is applicable. A distinction can be made between journeys which have clear start and end points [17] and those which can be considered open-ended [18]. Respective examples are an appeal for tax returns which is either granted or rejected at the end and the purchasing behaviour of a customer at a convenience store. The emphasis lies on the latter type of customer journey in this paper, where one is not certain if the actor will remain engaged in the process or not. In this case there is no clear endpoint to a customer journey. To circumvent this, a solution is adapted from [19] where a customer journey is considered finished if there has been a significant period of inactivity. Significance is here determined by a time period being longer than the 85th percentile of the periods of inactivity between events.

Taking this limitation into account, the following main statistics apply to the data used during the case study: There are 35060 cases which contain a total of 141510 events made up out of 271 possible activities that lead to 9127 different variants. In the studied environment, customers firstly buy a base product which allows them to install upgrades and expansions in the future. These further purchases are entirely optional and solely conducted based on the interest the customer has in the product. To give an idea on what information is helpful when using OARA, the relevant attributes from the case study are listed in Table I.

Attribute name	**Data type**
Customer ID	String
Installation time	DateTime
Event type	Integer
Product type	String
Product subtype	String
Number of products	Integer
Price	Integer

TABLE I: Table of the attributes in the case study dataset

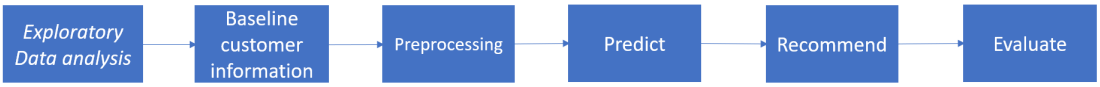

Fig. 2: A general overview of the components in the proposed approach.

In the scenario of this paper a relatively large number of the customer journeys end early, with 38% buying only the bare necessity and finding out that the product does not fit them well enough. A consequence of this is that almost all of the 9127 variants are then contained in the remaining 62% of the journeys, which makes them rather heterogeneous and leads to some difficulties in their predictions, as will be showcased in Section V.

Given the above circumstances, the aim is to strive towards the following two main goals:

1) Maximize metrics during the predictions of each of the events in the customer journeys. These predictions are done for future events in the customer journeys based on what has been observed in the past in them.
2) Maximize a customizable KPI during the recommendations. This KPI can be any value which is integral to the success of an organization in a specific area, such as sales or customer interaction. Examples of KPIs are the average purchase value, monthly sales bookings, and the customer engagement index.

Of a particular interest to this paper is to determine how useful it can be to take into account the inherent order which is contained in the customer journeys while achieving these goals. The reason for this is that from an intuitive standpoint this additional information should be helpful for doing the correct predictions and recommendations.

IV. OARA Overview

In Figure 2, a general overview can be observed which covers the main components of the system. The starting point is doing an optional exploratory data analysis to gain further insights into the dataset, for example by means of process mining. Afterwards, the baseline customer information is determined that form the base for future predictions and recommendations. Once the data to be used has been determined, some preprocessing is needed for the baseline customer information to reach its maximal potential during the next two phases, namely the predictions and recommendations. The predictions are used during the recommendations in OARA and as such these activities cannot be conducted in parallel once the preprocessing has finished.

A. Preprocessing

Preprocessing firstly involves the segregation of the set of customer journeys into smaller segments. The goal of this segregation is to allocate either specific types of customers which can be identified by domain knowledge, or customers which exhibit similar behaviour into their own groups. The reason for this is that then less behaviour needs to be kept in mind during the predictions and recommendations for a single group. As was mentioned in Section II, one way of segregating the customer used in this case study in this paper is based on the RFM values where customers who score similarly in each of the dimensions are grouped together. In the case study, this was done by calculating recency based on the installation time, frequency on the number of installations observed, and monetary value on the sum of all price values.

Another aspect of the preprocessing of OARA is obtaining the customer journey mapping by creating a process model as mentioned in the introduction. There are many scientific tools available to do this, most notably is ProM [20]. The main important point here is to strike a good balance between interpretability and complexity. The model should not be overly simplistic to the point where important parts of the customer journey are left out, while also making sure that the included events are common enough that they can be learned and predicted properly. The optional exploratory data analysis can help a great deal here, as this will aid in finding a proper balance since there is no ideal guideline to follow here. Upon having collected the process model, only those journeys which fit into the process model are used from the baseline customer information.

Apart from these tasks, the main interesting part of the preprocessing is the creation of representative customer journeys. The goal of these representative customer journeys is, as the name implies, to act as an artefact which represents most of the experiences encountered in the customer journeys between different customers. These are used during the recommendations, where they function as a sanity check to make sure that the recommended action is both optimal and reasonable. This is done by comparing how well the observed actions match between the representative customer journey and the journey which requires a recommendations. Their creation relies purely on the observed sequences of actions in the journeys.

Two types of representative customer journeys are proposed in this paper: the subset-based representative customer journeys (SRCJ) and the aggregated representative customer journeys (ARCJ). The choice between the two is determined based on whether a number of distinct customer journey variants can properly represent all users. If $Threshold \leq \sum_{i=1}^{n} weight(i)$ holds, the SRCJs are used, where n is the number of allowed variants and the weight is equal to the number of observed customer journeys that followed the same path as the variant. The SRCJs then simply become the n most frequently observed customer journey variants.

If on the other hand $Threshold > \sum_{i=1}^{n} weight(i)$, the ARCJs are used. For the creation of the ARCJs, all customer journeys are divided into groups based on how well they score

according to the KPI which is to be maximized. Then, inside each group $\sum_{j=1}^{m} \frac{\sum_{i=1}^{n} Action(ij)}{n}$ is calculated. Here n is once again the number of customer journeys, m is the number of possible actions observed in the journeys and $Action$ is a boolean value based on if the action was present inside the i'th journey.

As a final note, both the SRCJ and ARCJ can be incrementally updated based on new information to fit into a streaming setting. For the SRCJ one only needs to check if a different sequence has become more common than the current least common SRCJ, and at the ARCJ, the averages can be changed based on a new journey which has a similar KPI value. If desired, this can also be configured to give preference to the newer customer journeys to make sure recent trends are taken into account during recommendations.

B. Predictions

Predictions are conducted with regard to the next event(s) that occur in the customer journey. Note that the predictions also include the option of predicting a customer journey to end. This is primarily interesting for journeys which do not have a set ending point, as in that case one predicts at which point the customer loses interest in continuing their journey. This is an avenue usually left unexplored for recommendation systems, where the main focus lies on monitoring the events actually logged by the system. Adding such knowledge of a customer losing interest can be useful by for example sending them a special offer to rekindle their interest.

Based on the presented conditions, the process of conducting the predictions using OARA is explained in further detail here. As the name indicates, the order in which the events have occurred inside the customer journey is taken into account here. This means that the past is not considered to be a bag of unordered events such as for example in the OCuLaR algorithm. This added structure improves the distinguishability of the information used for predictions based on the assumption that customers who have followed the same trail in their customer journey, have a high likelihood of taking similar actions in the future as well.

An example of how taking the order into account can help is given in Figure 3. Here the order in which B and D are conducted has a high influence on what occurs after event E. If for example $\langle B, D, E\rangle$ is observed then the next event is F with a probability of 65%, while observing $\langle D, B, E\rangle$ lowers this to only 10%.

To properly do predictions for any sequence observed, OARA employs predictors for each of these sequences. As such, predictors need to be trained for all these sequences before proper predictions can be conducted for new customer journeys. This process is described in Algorithm 1. Here the sequence is the current path of a customer journey, e.g. $\langle A, B, C\rangle$. In that case there are two presequences to take into account, namely $\langle A\rangle$ and $\langle A, B\rangle$. For these presequences the next event is stored as well as the relevant features based on the available data. Once all customer journeys have been checked in this manner, a predictor is trained on the features

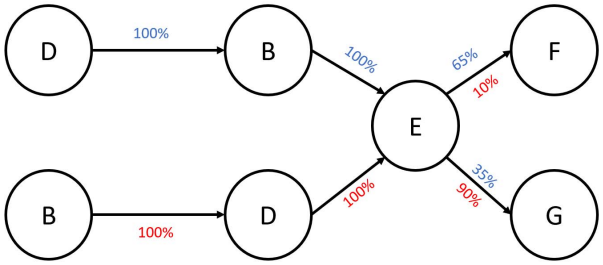

Fig. 3: Example of order influencing event probability.

and outcomes of the sequences in the base customer data. Note that in cases where the journeys contain a large number of heterogeneous events, it can be useful to not take into account the longer subsequences as there will be too little training information available for them to properly train the classifiers. Based on preliminary tests in our use cases, the performance will increase if the subsequence is shorter.

Algorithm 1 ObtainPredictors

Input: Training customer journeys n
Output: Predictors $predictorsArray$

```
Initialize sequencesArray, featuresArray,
  outcomesArray, predictorsArray
for i = 0 to len(n) do
    for j = 0 to len(sequence(i)) − 1 do
        presequence = first j events of sequence(i)
        if presequence not in sequencesArray then
            Add presequence to sequenesArray
        end if
        Obtain features(presequence) and add to
          featuresArray
        Add nextEvent(presequence) to outcomesArray
    end for
end for
for seq in sequencesArray do
    Fit predictor to featuresArray(seq) and
      outcomesArray(seq)
    Add predictor to predictorsArray
end for
return predictorsArray
```

Algorithm 2 OARA Prediction method

Input: Predictors P, Customer journeys n
Output: Predictions $predictionsArray$

```
Initialize sequencesArray, featuresArray, predictionsArray
for i = 0 to len(n) do
    Add sequence(i) to sequencesArray
    Obtain features(i) and add to featuresArray
end for
for j = 0 to len(sequencesArray) do
    Obtain prediction(j) based on predictor P[sequencesArray[j]] using featuresArray[j] and add prediction(j) to predictionsArray
end for
return predictionsArray
```

Once these predictors are obtained, predictions can be conducted on new samples. The outline of this is described in Algorithm 2. Here the current sequence of the new customer journeys is obtained as well as their features. Afterwards, the prediction is conducted based on the pre-learned predictor which was tailored towards that sequence to give an optimized prediction for the order of observed events.

Algorithm 3 OARA Recommendation method

Input: Representative journeys RCJ, Predictions P, Prediction Sequences PS, Conditions C
Output: Recommendations $recommendationsArray$

```
1:  Initialize distanceArray, recommendationsArray
2:  for i = 0 to len(P) do
3:      Initialize distancesArray
4:      for j = 0 to len(RCJ) do
5:          Obtain distance(i)(j) between PS(i) and RCJ(j)
6:          Append distance(i)(j) to distancesArray
7:      end for
8:      Initialize foundRecc = False
9:      for k = len(RCJ) to 0 do
10:         if foundRecc == False then
11:             currentDist = distancesArray(i)(k)
12:             if C based on currentDist are met then
13:                 Get recommendation based on P(i) and RJC(k) and add to recommendationsArray
14:                 foundRecc = True
15:             end if
16:         end if
17:     end for
18: end for
19: return recommendationsArray
```

C. Recommendations

The recommendations are aiming at maximizing a previously chosen KPI. This however does not mean that only the most profitable action is recommended for all customer journeys, as it is also taken into account how likely a customer is to take the recommended action. This is where the previously created representative customer journeys come into play. The representative journeys provide insights into which customer journeys led to higher KPI values and it is then possible to check how well a new customer journey aligns to the representative ones to get an idea of how reasonable a recommendation from that representative journey would be.

The general outline of how the recommendations using OARA are done is given in Algorithm 3. The two most interesting points here are the $distance$ and conditions (C) parameters used in respectively Lines 5-6 and 12. The $distance$ between the representative journey and the new customer journey is measured based on how well the events in the new customer journey match up with the representative journey. One way of doing this is using one-hot-encoding for all product types available at each of the observed events in the customer journey. This is the one we used in our case study. After that, the differences in value between the representative customer journey and the current customer journey can be calculated. It should be noted that this can be adapted to whatever preferred distance measure.

The other important parameter is the conditions (C), which specify the constraints to which a recommendation needs to adhere. These are based on the relative distance of a customer journey to the representative journey. An example of such conditions can be found in Table III. Here there are 4 tiers, where the actual values of the percentiles can for example be, based on the distances observed, between the representative journeys and all journeys in the baseline customer information. This creates a baseline for the distances on which new journeys can be judged. The exact conditions are tunable based on the context in which OARA is employed. In the example if a new journey falls into the 'Best' distance tier for a representative customer journey then the top 2 most likely events in the representative one are recommended. In the 'Good' distance tier, if any of the top 2 predictions for the new journey match with what occurs in the representative journey, then they are recommended. In the next tier the same holds for the topmost prediction. In case the journey is not similar to the representative one, the recommendation will not be based on it since the chance of following the recommendation will be too small.

These conditions are tested in the order from the highest ranked customer journey to the lowest ranked one, to try and route the customer on a path that maximizes the KPI.

Distance Tier	Distance%	Conditions
Best	0-15	Recommend 2 most likely events based on representative journey
Good	16-50	Recommend any of the top 2 predictions that match the representative journey
Decent	51-85	Recommend the top prediction if it matches the representative journey
Poor	86-100	Do not recommend based on this

TABLE II: Example of conditions on the recommendations

Furthermore it is never recommended for a customer to stop their journey as removing contact with the customer is only of use to an organization based on very specific conditions. In the unlikely event that none of the conditions can be met, the most often observed action from the highest scoring representative journey can always be recommended. This is done to still give some sort of advice which can lead to a KPI maximization.

V. Experimental Evaluation

As was previously mentioned, the evaluated customer journeys involved the initial purchase of a base product to which upgrades and expansions can be attached in the future. The customer journeys were then split into 8 groups based on if their *RFM* values were relatively high or low. In this section, the *HLL* and *HHH* groups are considered. These were chosen as they contained the most customer journeys while also being most interesting from a business perspective. The *HLL* group covers the customers who only recently started their journey and still have to determine if they appreciate the product and as such holds a lot of potential value if their interest can be retained. Conversely, the *HHH* group involves the 'best' customers which are currently still interested in the product and who already have purchased a relatively large number of products. This means also that if relevant products can be recommended to them they are likely to also be interested in those products. Process mining was used to extract customer journey maps out of these groups, which helped to filter out the journeys that are very hard to predict. The journey here consists of events which occur in a sequential order and those used for predictions are the 4th and 5th observed ones. Note that for the *HLL* group only the 4th is considered as the others do not exist in this group since the customer would then fall in a higher RFM group. Furthermore, the 3 first events are all part of the initial setup of the product of the customer, and are therefore not predicted.

Outside of these initial segregations, OARA was configured here to use 10 ARCJs as there was too much variance for a reasonable number of SRCJs to properly represent the customer base. Based on preliminary tests the best performing algorithm to use as predictors for each sequence was the Support Vector Machine, which has been used for multi-label classification with success in the past [21]. The conditions were the same as listed in Table II.

The predictions and recommendations on these customer journeys have been conducted in a multitude of ways to facilitate an overview of how different approaches were able to tackle this dataset. The competitors we used to assess OARA were the gradient boosting trees [22] and OCuLaR [13]. The reason gradient boosting trees were chosen is because they delivered the most promising results compared to other traditional methods. All scores have been obtained in a cross-validated manner while optimizing parameters for the methods based on the relevant metric.

The main metric of the comparison used here is an altered version of the F1-score, which is normally built up from precision and recall but here mean average precision is used instead of normal precision. The reason for this is to prevent punishing additional predictions in case the correct prediction was already conducted, as average precision only updates when recall changes. To indicate the different F1-score it is called Mean Averaged F1 (MAF1). The following formulas show exactly how it is built up:

$$Recall = \frac{\#CorrectPredictions}{\#Items} \tag{1}$$

$$Precision = \frac{\#CorrectPredictions}{\#Predictions} \tag{2}$$

$$AP = \sum_{i=1}^{n} Precision(i)\Delta Recall(i) \tag{3}$$

$$MAP = \sum_{i=1}^{n} AP(i)/n \tag{4}$$

$$MAF1 = 2 * \frac{MAP*Recall}{MAP+Recall} \tag{5}$$

A. Predicting the next event

The main objective of the predictions is determining the very next event in the customer journey. For this reason, the MAF1 score has been obtained for the 5 top predictions for each of the 3 predictors, which can be found in Figures 4-6.

1) HLL Event 4: Based on the MAF1 scores in Figure 4, the predictions for this event are done mostly equally well for both OARA and Gradient-boosting, although OARA achieves a better initial prediction. As such, recall rises a bit faster using Gradient-Boosting, while OARA relies more on its initial high precision. OCuLaR performs worse here mostly due to there being a relatively large number of people who stop their customer journey at this event, which is a bit troublesome for it to identify. This is due to missing the feature information which the other two competitors have access to.

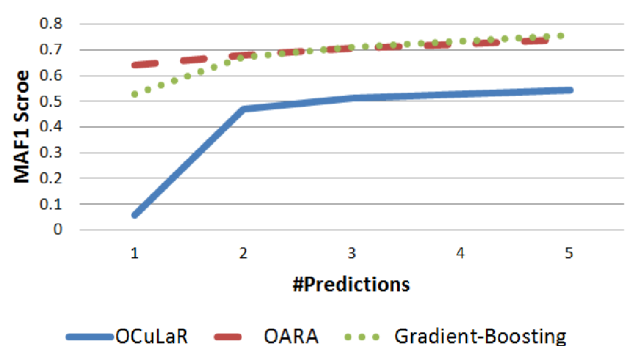

Fig. 4: MAF1 scores for the 4th event in group *HLL*.

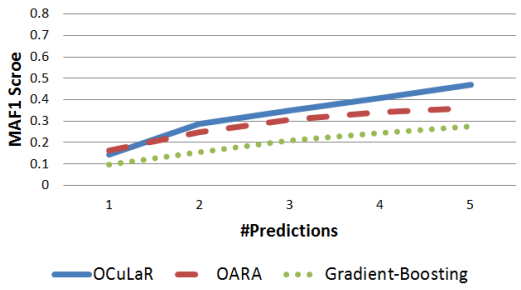

Fig. 5: MAF1 scores for the 4th event in group *HHH*.

2) HHH Event 4: Based on the performance in Figure 5, all algorithms perform poorly here. This is caused by the large number of options to choose from which are all relatively uncommon and, there is an insufficient amount of information to distinguish between them. This shortage of information allows OCuLaR to outperform the other two methods due to being more effective in obtaining insights based on just the products bought if multiple predictions are allowed. OARA then, thanks to taking the order into account, still outperforms Gradient-Boosting, but is mostly less effective than OCuLaR.

3) HHH Event 5: When comparing Figures 5 and 6, the prediction quality and conditions are about equal. However, this time OARA is able to outperform the other two methods instead of OCuLaR. The additional event has led to enough information becoming available that the combination of the order and features has become well-suited to do the predictions.

B. Using a span

A *span* here refers to a timespan during which it is allowed for a prediction to be valid. To clarify, if *span* = 3 then if the predicted action shows up either in the next event, the event after that, or the event following that then the prediction is considered to be correct. This can be useful when one is relatively sure that a group of actions will be conducted in the near future without the order being set in stone. An example of this is a user of an online music service who has already bought 3 albums of a single artist, where one can be relatively certain they will buy another album of that artist but not which one. Usage of a *span* for sequences of events is

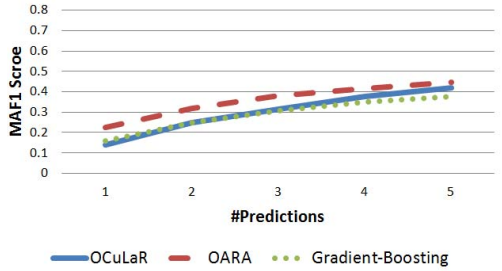

Fig. 6: MAF1 scores for the 5th event in group *HHH*.

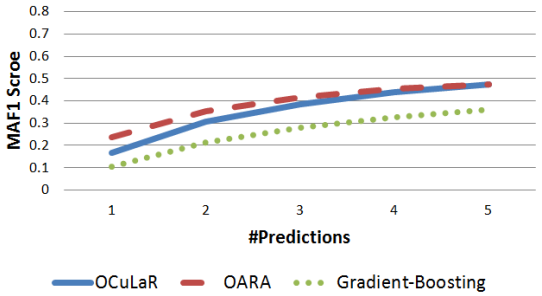

Fig. 7: MAF1 scores for the 4th event in group *HHH* with span of 3.

not unprecedented and has been used with success in the past in [23] which is adapted to the current situation.

The effect of using the *span* for the predictors here is exemplified in Figure 7, where the effect was most noticeable. A *span* of 3 was used here and when compared to the $MAF1$ scores obtained in Figure 5, the $MAF1$ scores here are higher due to the relaxed prediction conditions, as is to be expected. However, not all predictors profit equally from this and it allows for OARA to now outperform OCuLaR while with a *span* of 1 this is reversed. This shows that given a scenario where a span is reasonable using OARA can help improve the quality of the predictions.

C. Including additional context information

From an intuitive standpoint, it makes sense that to increase the predictive qualities it is helpful to include additional information. This enriches the customer journey by providing additional context to the observed events, much like taking the order into account did. To test this hypothesis, an additional dataset has been obtained and deployed in the use case. The comparison of the results with and without the added dataset can be found in Table III.

In this table the $MAF1$ score is given for the predictions of the events when allowing for 1 or 5 predictions without using a larger span. Based on the $MAF1$-scores the main improvements are found at the prediction of event 5 of the *HHH* group, while when allowing for multiple predictions the $MAF1$-scores of the predictions for the 4th event in both groups also seems to rise. The only time when there is little effect is during the prediction of the 6th event in the *HHH* group. As such the context data used here is mainly of use during the earlier stages of the customer journey. Additional context information from a different source may prove useful to improve on the later predictions. This experiment has shown that providing additional context information can lead to an increase in metrics when using OARA.

D. Recommendations evaluation on KPIs

The evaluation of the recommendations for the next event in the customer journey is need to be done in a novel way. The evaluation of recommender systems is traditionally based on measures related to the precision of the recommendations

TABLE III: Comparing MAF1 scores based on the presence of context data

	HLL-4@1	HLL-4@5	HHH-4@1	HHH-4@5	HHH-5@1	HHH-5@5	HHH-6@1	HHH-6@5
No context data	0.642	0.741	0.161	0.360	0.223	0.447	0.388	0.582
With context data	0.652	0.783	0.146	0.415	0.291	0.479	0.382	0.583

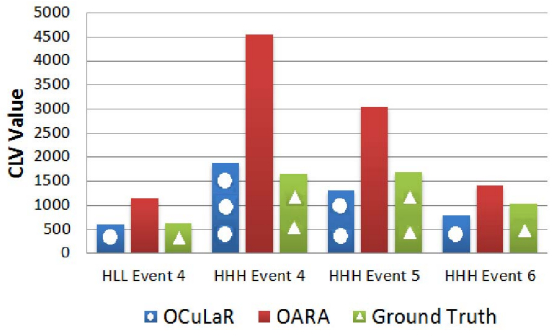

Fig. 8: Comparison of the CLV values on recommendations.

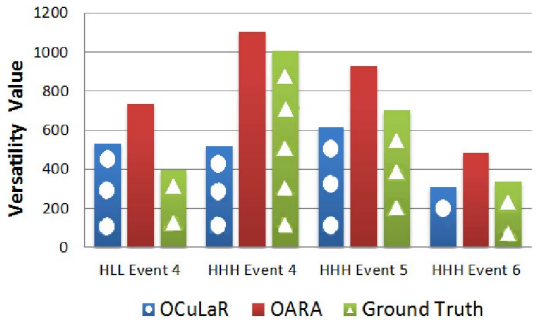

Fig. 9: Comparison of the versatility values on recommendations.

and not the maximization of a KPI. To solve this the following metric is introduced, which aims to capture how much value the recommender brings in terms of the KPI:

$$TotalKPI = \sum_{i=1}^{n} : KPI(recc(i)) \tag{6}$$

Here n is the number of customers which have been recommended a next step in the journey, and the KPI is calculated based on a specific recommender system. Note that this operates under the restriction that a KPI is used which can be calculated at any point in the customer journey. In other words, every step contributes a certain amount of KPI value. If this is adhered to, then *TotalKPI* allows for an estimation of the recommendation's effect on the KPI. This however also relies on the assumption that the customers always follow the recommendation. It should be noted that this assumption is different from what one can expect to see in real life, and is mostly in place due to the lack of any prior research on how often recommendations are followed up on by customers. In case one wishes to be more realistic then one can for example assume only half of the recommended events are followed by the customers, while taking the KPI from the ground truth in the remaining cases.

$TotalKPI$ was calculated for OARA under the positive assumption that all recommendations are followed as well as OCuLaR and the ground truth to see if there is any positive effect. The first KPI used here was the Customer Lifetime Value(CLV), a KPI which aims to capture how valuable a customer is to an organization. Based on the specification in [24], the CLV is here non-contractual, i.e. customer can always stop buying products. It is also dynamic, i.e. each action has an effect on the KPI. The CLV is for this example solely depends on the revenue per step in the customer journey.

In Figure 8, the CLV values have been calculated for 4 recommendations. The first two recommendations involving the *HHH* group have higher CLV values for all 3 recommendations methods due to having more samples increasing the overall CLV. For each of the 4 recommendations, OARA is outperforming the alternatives, which is caused by only OARA optimizing explicitly for the CLV value. There are also times when customers strictly following the OCuLaR recommendation would have a lower CLV value than the ground truth (i.e. their real behaviour). The same can however be said for OARA, which aims to have a more direct influence by giving a recommendation that immediately increases the KPI. The sum of CLV values contributed by OARA may generally not be high enough, yet it will not sink below the ground truth.

Another KPI which can be relevant is the versatility in the types of products a customer purchases. This is then measured in the number of different types a customer purchases during their journey, and as such the highest ranked representative customer journeys are those which involve the purchase of many different types of products. The result when using this as the KPI when calculating $TotalKPI$ can be seen in Figure 9. The result is similar to Figure 8 in the sense that OARA outperforms both OCuLaR and the ground truth, although this time with slimmer margins. Most notably here at the 4th event of group *HHH*: the ground truth is very close to the OARA recommendation which is caused by this group of customers naturally buying many different items already. Of course, this is to be expected in this golden category of customers (*HHH*), however, we still see an increase in different KPI even for this category. Recommendations here are mainly much more powerful for customers with much more conservative choices in their initial purchasing behaviour.

VI. Conclusion and Future Work

In this paper we presented OARA, an Order-Aware Recommendation Approach that focusses on predicting next

bahviours in the customer journey, and recommending the ones that maximize any organization-specific Key performance indicator (KPI). The approach proposed in this paper allows for the predictions and recommendations on datasets which fit the concept of a customer journey. We have showcased that one can go beyond merely visualizing the journey in a process model by utilizing the model for these tasks. The additional novelty we contributed in the work was investing of the positive effects of taking into account the order of events in the previous parts of the journey. These are observable during the predictions both in situations when exactly the very next behaviour must be predicted or when slightly more remote future actions are of interest. It was also shown that OARA can be further enriched by effective use of an additional source of information. These predictions are then used in combination with the representative customer journeys during the recommendations to find a recommendation that both increases the KPI and is well suited based on the actions previously observed in the customer journey.

Interesting future work includes looking further into further evaluation metrics in settings where recommendations are aimed at improving a *general* KPI. The main shortcoming currently lies in the assumption that the customers follow recommendations blindly, which was put into place due to a lack of access on how often customers actually follow the given recommendation. This can be achieved using an appropriate A/B testing [25] for instance. As such, a case study of the effectiveness of recommendation could provide a lot of value to the assessment of recommender systems. Furthermore OARA has currently only been employed in a single scenario. Therefore deploying it in a different environment will likely lead to further insights on optimizations and generalizations in areas which were not significant in the current one. It is preferable that the new scenario includes structures where events can be conducted in parallel, after which a specific event follows. One can in this case apply scenarios from parallel sequential pattern mining like [26]. OARA can handle such a sequence of events perfectly given the awareness of past behaviour in the customer journey, however no such patterns exist in the data of the current case study. Another interesting research direction that we want to investigate in the future is investigating the possibilities of online conformance checking [27] in investigating, in the real-time, deviations from dynamically evolving KPIs. This will enable for instance switching between a set of different KPIs as the journey evolves.

REFERENCES

[1] Katherine N Lemon and Peter C Verhoef. Understanding customer experience throughout the customer journey. *Journal of Marketing*, 80(6):69–96, 2016.

[2] Julia Wolny and Nipawan Charoensuksai. Mapping customer journeys in multichannel decision-making. *Journal of Direct, Data and Digital Marketing Practice*, 15(4):317–326, 2014.

[3] Alessandro Terragni and Marwan Hassani. Analyzing customer journey with process mining: From discovery to recommendations. In *6th IEEE International Conference on Future Internet of Things and Cloud, FiCloud 2018, Barcelona, Spain, August 6-8, 2018*, pages 224–229, 2018.

[4] Gaël Bernard and Periklis Andritsos. A process mining based model for customer journey mapping. In *Proceedings of the Forum and Doctoral Consortium Papers Presented at the 29th International Conference on Advanced Information Systems Engineering (CAiSE 2017)*, pages 49–56, 2017.

[5] Guy Shani and Asela Gunawardana. Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer, 2011.

[6] Asbjørn Følstad and Knut Kvale. Customer journeys: a systematic literature review. *Journal of Service Theory and Practice*, 28(2):196–227, 2018.

[7] Anna Meroni and Daniela Sangiorgi. *Design for services*, pages 83–85. Routledge, 2016.

[8] Marc Stickdorn, Jakob Schneider, Kate Andrews, and Adam Lawrence. *This is service design thinking: Basics, tools, cases*, volume 1, pages 17–36. Wiley Hoboken, NJ, 2011.

[9] Stefan Holmlid and Shelley Evenson. Bringing service design to service sciences, management and engineering. In *Service science, management and engineering education for the 21st century*, pages 341–345. Springer, 2008.

[10] Wil Van Der Aalst, Arya Adriansyah, Ana Karla Alves De Medeiros, Franco Arcieri, Thomas Baier, Tobias Blickle, Jagadeesh Chandra Bose, Peter van den Brand, Ronald Brandtjen, Joos Buijs, et al. Process mining manifesto. In *International Conference on Business Process Management*, pages 169–194. Springer, 2011.

[11] Boudewijn Frans van Dongen. Process mining and verification, 2007.

[12] Nauman Chaudhry, Kevin Shaw, and Mahdi Abdelguerfi. *Stream data management*, volume 30. Springer Science & Business Media, 2006.

[13] Reinhard Heckel, Michail Vlachos, Thomas Parnell, and Celestine Dünner. Scalable and interpretable product recommendations via overlapping co-clustering. In *Data Engineering (ICDE), 2017 IEEE 33rd International Conference on*, pages 1033–1044. IEEE, 2017.

[14] Jan Roelf Bult and Tom Wansbeek. Optimal selection for direct mail. *Marketing Science*, 14(4):378–394, 1995.

[15] Mahboubeh Khajvand, Kiyana Zolfaghar, Sarah Ashoori, and Somayeh Alizadeh. Estimating customer lifetime value based on RFM analysis of customer purchase behavior: Case study. *Procedia Computer Science*, 3:57–63, 2011.

[16] Derya Birant. Data mining using RFM analysis. In *Knowledge-oriented applications in data mining*, pages 91–108. InTech, 2011.

[17] Morris Foster, Susan Whittle, Stuart Smith, and Peter Hyde. Improving the service quality chain. *Managing Service Quality: An International Journal*, 1(1):41–46, 1991.

[18] Mirela Elena Nichita, Marcel Vulpoi, and Georgiana Toader. Knowledge management and customer relationship management for accounting services companies. *Chinese Business Review*, 12(6):435–442, 2013.

[19] Marwan Hassani, Sergio Siccha, Florian Richter, and Thomas Seidl. Efficient process discovery from event streams using sequential pattern mining. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 1366–1373. IEEE, 2015.

[20] Boudewijn F Van Dongen, Ana Karla A de Medeiros, HMW Verbeek, AJMM Weijters, and Wil MP Van Der Aalst. The ProM framework: A new era in process mining tool support. In *International conference on application and theory of petri nets*, pages 444–454. Springer, 2005.

[21] Glenn M Fung and Olvi L Mangasarian. Multicategory proximal support vector machine classifiers. *Machine learning*, 59(1-2):77–97, 2005.

[22] Ping Li, Qiang Wu, and Christopher J Burges. Mcrank: Learning to rank using multiple classification and gradient boosting. In *Advances in neural information processing systems*, pages 897–904, 2008.

[23] Marwan Hassani, Daniel Töws, Alfredo Cuzzocrea, and Thomas Seidl. BFSPMiner: an effective and efficient batch-free algorithm for mining sequential patterns over data streams. *International Journal of Data Science and Analytics*, pages 1–17, 2017.

[24] Siddarth S Singh and Dipak Jain. Measuring customer lifetime value: models and analysis. *SSRN*, pages 10–17, 2013.

[25] Ron Kohavi and Roger Longbotham. *Online Controlled Experiments and A/B Testing*, pages 922–929. Springer US, Boston, MA, 2017.

[26] Marwan Hassani and Thomas Seidl. Towards a Mobile Health Context Prediction: Sequential Pattern Mining in Multiple Streams. In *MDM*, pages 55–57. IEEE, 2011.

[27] Sebastiaan J. van Zelst, Alfredo Bolt, Marwan Hassani, Boudewijn F. van Dongen, and Wil M. P. van der Aalst. Online conformance checking: relating event streams to process models using prefix-alignments. *International Journal of Data Science and Analytics*, Oct 2017.