Predicting Activities of Interest in the Remainder of Customer Journeys Under Online Settings

Lisan Wolters and Marwan Hassani[0000−0002−4027−4351]

Eindhoven University of Technology, The Netherlands
lisanwolters@outlook.com, m.hassani@tue.nl

Abstract. Customer journey analysis is important for organizations to get to know as much as possible about the main behavior of their customers. This provides the basis to improve the customer experience within their organization. This paper addresses the problem of predicting the occurrence of a certain activity of interest in the remainder of the customer journey that follows the occurrence of another specific activity. For this, we propose the HIAP framework which uses process mining techniques to analyze customer journeys. Different prediction models are researched to investigate which model is most suitable for high importance activity prediction. Furthermore the effect of using a sliding window or landmark model for (re)training a model is investigated. The framework is evaluated using a health insurance real dataset and a benchmark data set. The efficiency and prediction quality results highlight the usefulness of the framework under various realistic online business settings.

Keywords: Process Mining · Process Prediction · Customer Journey Analysis · Streaming Data · Machine Learning · Deep Learning

1 Introduction

Customer journey analysis is useful for companies trying to understand how the customer interacts with the company. Next to understanding the customer journey it can also be used to improve the customer experience [1]. Customers can interact with a company over multiple channels, such as website visits, phone calls, physical presence at stores, etc. Not all interactions (or touchpoints) provide the same customer experience and satisfaction [2]. Next to understanding current customer journeys, it is also interesting for companies to predict whether customers will interact with a certain touchpoint on a later moment in their journey. Knowing in advance which customer will encounter certain touchpoints, might provide the option to prevent the occurrence of touchpoints that often indicate a negative feeling towards the journey which in turn might result with saving resources. Current research has already shown interest in next event prediction and final outcome prediction for running customer cases [3, 6]. In this paper, the research conducted will investigate whether the customer will interact with a certain activity in the remainder of its journey. Therefore, neither next activity prediction nor final outcome prediction will alone be sufficient. Filling the gap in future touchpoint of interest prediction is achieved by providing a repeatable framework for future high importance activity prediction (HIAP). The main use case of this work comes from a health insurer company that we will refer to as Yhealth. Yhealth wants to retrieve insights in which customers are most likely to call Yhealth. Performing a call is often experienced bad; therefore, it is interesting to prevent such interactions. A first step in prevention is knowing which customer will call. For this purpose, a data set containing declaration data of the customer is provided. The goal is to predict at a certain moment in the customer journey which customers will call Yhealth in the remainder of their journey. The solution proposed in this paper uses online process mining techniques to analyze the current customer journeys. The insights gathered serve as basis to indicate the decision moment (DeM) and potential activity (PoAc). For customer journeys reaching the DeM it should be predicted whether the customer will interact with the PoAc in the remainder of its journey. Machine and deep learning models are trained to perform predictions. The solution provides a repeatable framework to predict the occurrence of a PoAc in a customer journey. The performance of the different prediction models is evaluated. Next, research is conducted in the resources needed to keep a model up to date to recent customer journeys with respect to the quality gain, using online settings with a sliding window model and a landmark window model. This research shows that it is important to focus on recent traces to observe and react on changing behaviour of the customer.

The paper is structured as follows: Section 2 provides an overview of related work. Section 3 contains notations used in the paper and explains the research problem in more details. Section 4 defines the proposed framework, which is then evaluated in Section 5. Section 6 concludes the paper with an outlook.

2 Related Work

Predicting next events and timestamps in a running trace is discussed in several works. Though none of these works have the same assumptions, data or goal. In [7] a technique to analyze and optimize the customer journey by applying process mining and sequence-aware recommendations is proposed. These techniques are used to optimize key performance indicators to improve the customer journey by providing personalized recommendations. The goal of predicting what a customer will like differs from predicting what a customer will do. Especially predicting whether a customer will encounter an action that is often experienced badly is a different goal. Therefore, the second phase of sequence-aware recommendations is not applicable in the current context. Predicting a next event and its associated timestamp in a customer journey is discussed in [6]. They propose a RNN with the LSTM architecture for both next event and suffix prediction. Suffix prediction is applied by iteratively predicting the next event. This may result in a poor suffix quality as an error in a previous prediction is propagated to the next prediction. This approach encounters difficulties with traces in which the same event occurs multiple times as in that case the model will predict overly long sequences of that event. In the case of a health insurer, some events are expected to reoccur. Therefore, a solution for this limitation should be implemented to be applicable in the current context. Another approach on suffix prediction is applied in [5] by using an encoder-decoder GAN. The encoder and decoder are both represented by an LSTM network and allow the creation of variable length suffixes. This technique is used to predict suffixes up to the end of the trace. Furthermore, a suffix is not generated at a certain point in the customer journey, which is of high importance in the current research. Different ML and DL techniques for outcome predictions are evaluated in [4]. The outcome of a trace is predicted for a journey up to x events in which x has values from 1 to 10. Predicting the final outcome of a case is not the same as predicting whether a certain activity will occur. However, it should be possible to adapt the final outcome to high importance activity prediction. But nonetheless the technique is not applicable in the current research as a prediction should be provided as soon as a certain proposition holds for that trace, instead of after x events. In [8], the authors propose a framework for online prediction of the final outcome of retailer consumer behaviour using several aggregation methods.

3 Problem Exposition

This section defines the notation needed to understand the HIAP framework and describes the research problem in more details. Let $\mathcal{CJ} = (cj_1, cj_2, ..., cj_n)$ be a log containing the customer interactions. Each row in the log $cj_r = (cu_j, t, i, ia_1, ..ia_m)$ defines a single interaction of customer cu_j. The customer conducted touchpoint i at time t. The interaction of the customer may have interaction attributes $(ia_1, ...ia_m)$. Later, $\mathcal{CJ}$ is converted into an event log. Let $\mathcal{L} = (e_1, e_2, ...e_n)$ be the event log of the customer journey. Each row in the log $e_r = (c_i, t, a, d_1, ..d_k)$ defines a single event performed by one case identifier c_i. Each customer cu_j can be mapped to a c_i. The touchpoint of the interaction of the customer is renamed to an activity a and the activity is performed at time t. Each touchpoint i will be mapped to an activity a, but multiple touchpoints might be mapped to the same activity a. Furthermore, events can have attributes $d_1, ...d_k$, extracted from the interaction attributes. The log $\mathcal{L}$ contains all traces of the customers in CJ. Let $\sigma_i =< e_1, e_2, ..., e_{|\sigma_i|} >$ define the trace of case identifier c_i. The α-prefix is the trace up to and including the first α events. The suffix is defined as event $(\alpha + 1)$ until the end of the trace.

This work aims to use process mining techniques to improve customer journey analysis and use the insights to improve the customer experience. A repeatable framework for future touchpoint prediction in a customer journey is proposed. The result can be used to make the customer journey smoother, which will result in a more satisfied customer. For a customer journey a PoAc and DeM in the trace will be defined. Based on DeM x, we know the x-prefix $< e_1, ...e_x >$ of a customer journey. Using the information in the x-prefix, the goal is to predict whether PoAc y will occur in the x-suffix of the customer. Where the x-suffix is $< e_{x+1}, ..., e_{|trace|} >$. Customer journeys may change rapidly, therefore the prediction models should facilitate updates and the framework is tested by means of data streams. Using data streams stresses the effect of incorporating changing behaviour of customers. Without using data streams, models are based on older customer data and in the case of changing customer behaviour the prediction will become unreliable. When models are retrained over time the recent changes in customer behaviour is still considered and models will provide predictions with a higher performance.

4 High Importance Activity Prediction Framework

This chapter introduces the high importance activity prediction (HIAP) framework to predict the occurrence of an interesting touchpoint in the remainder of the customer journey based on the journey up to a specific point in time. The prediction uses information of the event log prefixes and possible customer information to predict for a specific customer whether (s)he will have a specific interaction in the future. Figure 1 shows an overview of the framework. This chapter explains the steps of the framework.

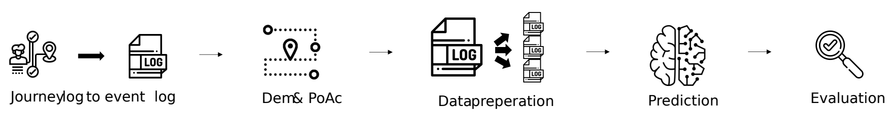

Fig. 1. Schematic overview of the proposed framework

The goal of the first step is to create a preprocessed event log $\mathcal{L}$ that can be used for the research. Preprocessing is needed to combine data of different sources, infer missing data and remove unnecessary data [9]. Different scenarios require different data harmonization techniques. Examples are data cleaning, transforming interactions and transforming a customer journey to an event log.

4.1 Critical Moments

The process model of event log $\mathcal{L}$ is used for defining critical moments. The critical moments are the decision moment(DeM) and the potential activity (PoAc).

Decision moment definition The goal of HIAP is to predict whether a certain activity will occur based on a predefined moment in the trace. This specific moment can be defined either by a specific activity or by a proposition based on the events in the trace. The first time that such an activity occurs or the proposition holds will be taken as the DeM of the trace. When determining the DeM two criteria should be considered. First, the goal of the prediction is to be able to adjust the remainder of the trace and prevent the occurrence of a certain activity or to be able to save resources. As a result, the prediction should be early in the process. Second, the prediction should be as accurate as possible.

In general, more accurate predictions can be provided at the moment that more information is available about the current process. Therefore, a balance should be found between choosing an early DeM and the quality of the prediction [10]. As the prediction takes place at a certain moment, only the traces in $\mathcal{L}$ that at some moment satisfy the condition of a DeM should be considered. The traces that do not satisfy the DeM should be removed from the log.

Potential activity definition The PoAc is the activity of which it is preferred to know whether it will occur in the remainder of the customer journey. The DeM should be a proposition that is met earlier in the trace than that the PoAc occurs. However, the PoAc may be an activity that is occurring at a random moment in the suffix of the trace with respect to the DeM.

4.2 Data Preparation

Prior to the prediction phase a training, a validation and a test set should be created. Two methods are used to create those sets, one being a static method and the second method a streaming setting. Method one uses chronological in time the first 70% of the data as training data, the next 10% as validation data and the last 20% as test data. For the second method, a sliding window and a landmark model are used to investigate the effect on the training time and prediction performance. These results provide insights in the need to use all historical data or only recent historical data to keep the prediction models up to date. Using a wider period of time results in a considering more customer journeys and more likely a wider spectrum of use cases. While narrowing the time window provides a more detailed focus on recent customer journeys and provides more details on recent behaviour. In this case, a start date and end date of the window is defined. The training and the validation sets are composed of the traces that are completed in this window. The test set is constructed of the set of traces of which the proposition defining the DeM is satisfied in this time window, but that are not yet completed.

4.3 Prediction of the Potential Activity

In this paper three models are considered for the prediction of the PoAc to determine which model is most suited. The possible methods for prediction are not limited to these models; therefore, it is possible to consider other models too.

Random Forest Classifier In order to train a RFC, the traces first need to be represented as a set of features [**?**]. These features consist of a set of independent variables and one dependent variable. This set of independent variables should be deduced from the trace that is available up to the DeM as well as available customer details. The dependent variable represents the occurrence of the PoAc in the suffix of the trace. Resulting in a binary decision.

Long-Short-Term-Memory network The LSTM used in the research is inspired on the implementation of [4] for final outcome prediction. Their preprocessing entails multiple steps. First, they defined the number x of events which should be considered while creating the feature vector. The feature vectors only entail information of the event and trace attributes that are available up to that moment of the trace. The traces that did not contain at least x events are removed from the log. Last, the label indicating the final outcome of the current trace is assigned to the feature vector. This part of the feature vector is used to compare the prediction with the ground truth and to train model parameters. This preprocessing is not directly applicable in the current research. The event number of the DeM may differ from one trace to another, but for each trace the prediction should be provided at the DeM. For each trace, the number of events prior to the DeM can be extracted. Furthermore, a number y of events is defined, defining the preferred prefix length for each trace. Traces containing more than y events up to the DeM, should be shortened. Only the last y events up to the DeM should be kept. Traces that have less than y events up to the DeM should lengthened with artificial events, added to the start of the trace. The events occurring later than the DeM, should still be preserved. The trace suffix will be used to determine the dependent variable, indicating whether the PoAc occurs. The feature vectors are used as input to a LSTM network classifier. The model is trained with a two-stage learning strategy as explained in [4].

Generative Adversarial Network The GAN described is an adaption of the model in [5] for suffix prediction. The implementation needed some modification regarding the creation of the training, the validation and the test set and the number of prefix and suffixes created for each trace. [5] created the training, the validation and the test sets by randomly selecting instances from the complete log. In this research those are defined based on the timestamp of the DeM or based on the timeframe. Secondly, one prefix-suffix combination should be created per trace based on the DeM. The PoAc activity prediction could be determined by the occurrence of the PoAc in the suffix returned by the model.

4.4 Model Comparison and Future Model Use

The next step is to evaluate the performance of each classifier to judge the trustworthiness of the classifiers and to compare the different models. Depending on the research field and goal of the research the quality of each model will be accessed by the F1-score and/or recall. Generally, a higher score implies that the model is outperforming the other models [9]. Furthermore, the three models should also be compared to a baseline model. As baseline model a random predictor is used. The random predictor uses the distribution of the occurrence of the PoAc in the training set and predicts for the test set that the PoAc will occur in the same percentage of cases. The average prediction performance over 1000 runs is used as result for the baseline model. After training a model, the goal is to predict for new cases, as soon as the DeM property holds, whether the PoAc will occur. Predictions should be as reliable as possible in such cases; therefore, the model that is expected to be most trustworthy should be used.

The model that is evaluated to be the best model, can be used for future instance predictions. After training a model, the model can be stored, such that the model can be used for future predictions of the PoAc. At the moment that the proposition defining the DeM holds for a new customer journey, it can be represented with the same feature representation as the original data. A prediction on the occurrence of the PoAc will be provided by the model. The prediction can be used to act upon to improve the customer experience.

5 Experimental Evaluation

This section evaluates the application of the HIAP research on the Yhealth and benchmark BPI 2012 dataset.

5.1 Health Insurer Data Set

The Dutch health insurer data set contains details about the declaration process for customers. The log $\mathcal{CJ}$ covers a time period of two months, recording for all interactions cj_r the touchpoint i, its timestamp t and the customer identifier cu_j. In addition, anonymized customer details are available and touchpoints are related to further attributes, for example for a call the question is recorded. The data harmonization is conducted with the help of Yhealth. Steps taken are filtering of phone calls based on the subject, mapping of touchpoints to belong to a declaration and filtering incomplete traces. This resulted in an event log $\mathcal{L}$ consisting of $95,457$ traces accounting for nearly $400,000$ events. Most traces are relatively short as 95% of the traces had less than 10 events. The goal for Yhealth is to determine whether a customer will call as a follow-up to obtaining the result of the declaration. Calling is often perceived negatively by the customer; therefore, Yhealth would like to prevent the occurrence of a call. The first step to prevent the call is to know who will call. For that reason, the PoAc is defined as a call. The DeM is the moment that the result of a declaration is sent to the customer. This moment is chosen as earlier in the trace, for a lot of traces not enough information is available for the prediction and the result of the declaration will provide valuable information for the prediction. The log is imbalanced, as only 3.5% of the traces contain a call event on a later moment than receiving the result on a declaration. The set is used to create a training, a validation and a test set. The training set is undersampled such that the occurrence of the PoAc is more evenly distributed in the suffix with respect to the DeM.

The next step is to convert the traces to input for the RFC, LSTM and GAN. For all three models the traces up to the moment that the customer receives the results of a declaration is used as input to train a model for predictions. In the case of the RFC the traces have to be converted in a set of independent decision variables and one dependent variable which is the PoAc. The independent variables contain information of trace and event attributes. The input features of the LSTM network contain information on trace attributes and event attributes. All input features should have the same length; therefore, each trace is preprocessed such that it contains 5 events up to the DeM. The preprocessing of [4] is used to create the feature representation. The GAN network uses the original prefixes up to the DeM. The input of the training set also contains the suffixes which are either the suffix up to the PoAc or the complete suffix when PoAc is not in the suffix. The feature representation as proposed in [7] is the input for the encoder-decoder GAN.

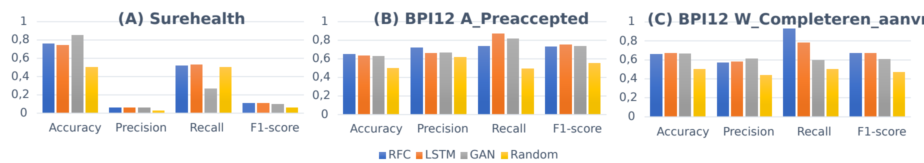

Fig. 2. Performance measures of the prediction models in offline setting

To determine which model can be used best, the three models and baseline model should be compared. The result is shown in Figure 2(A). None of the models is performing best on all four performance measures. For the case of Yhealth it is most important to know whether a customer is likely to call. Therefore recall together with F1-score are the most import performance measures. On these two measures the LSTM and RFC model are performing best. The F1 score of both these models is doubled with respect to the random classifier; therefore, outperforming the baseline model. Without affecting the quality of the model, LSTM networks usually require a higher hardware requirement to train and use the model [4], which is not always available. Furthermore, RFC models are easier to understand and explain for humans. Accordingly, the RFC might be selected as the best prediction model for Yhealth.

Next to comparing the three prediction models on the complete data set, research is conducted in applying a sliding window and landmark model. A sliding window model only trains over the most recent instances, while a landmark model trains on the complete history of available data. Therefore, it is expected that a landmark model needs more resources to train a model. However, it is also expected that the quality of the predictions will be higher, as more training data is available. For this purpose, sub windows of the complete data are used to create the training, the validation and the test set for the sliding window and landmark model. In the current research, all models are trained on a CPU. If a GPU would be available the models could benefit from improved parallel computations. A GAN and LSTM network are expected to benefit more from a GPU, while the RFC is expected to be faster on a CPU.

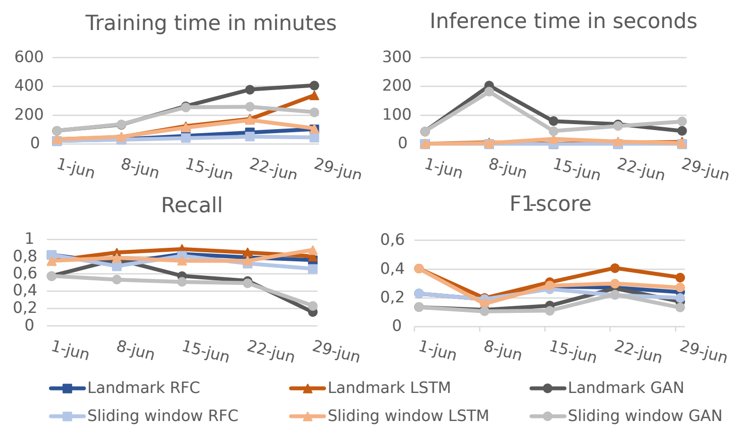

Fig. 3. Sliding window and landmark model results for the Yhealth data set.

The results of using a sliding window and landmark model on the Yhealth data set is shown in Figure 3. For the sliding window, each window contains two weeks of data and the window shifts with one week for each new window. For the landmark model the first window contains two weeks of data, the next windows each increase with the data of one extra week. As can be seen in Figure 3 the training time of the landmark model increases as the window size increases. Considering the same window, the RFC model is trained faster than the LSTM model and the LSTM models is faster than the GAN model. The inference time for the RFC and LSTM model is similar, but the GAN model is slower. For each model, the precision and F1-score of the landmark model is at least as high as their counterparts of the sliding window. This shows that the landmark model is eager to learn using more journeys, even if these journey are already a little older. Considering the recall score, up to the window ending at June 22, the landmark model is performing better than the sliding window for each prediction model. For the window ending at June 29, the GAN and the LSTM models trained over the sliding window have higher recall scores than the two models trained over the landmark model. The GAN model is performing worst for almost all windows. As of the window ending at June 15, the LSTM model on the sliding window and landmark model scores are slightly higher than the RFC model. However, the training time of the RFC model is considerably shorter. For a model to provide predictions, it is important to regularly update the model to new instances. Updating a model is easiest if training takes as less time as possible, but the results should not be affected by the reduction of the training time. Especially the time to train the LSTM model on the landmark model is too long for the last window. Therefore, the landmark LSTM model is not the most preferred model. The gain in performance for the LSTM sliding window model is small with respect to the RFC model on the landmark model. The performance of the RFC model with the sliding window is again slightly lower.

However, the model with the lowest performance requires the shortest training time. As the RFC model is easier to understand, the RFC model is the preferred model to use. The running time of the landmark model of the RFC is not yet too long; therefore, the landmark model is preferred over the sliding window.

5.2 BPI 2012 Data Set

Since the data of Yhealth is confidential, the HIAP framework is replicated on the public available BPI 2012 event log. The BPI 2012 challenge event log contains data of the application process for a personal loan or overdraft within a Dutch financial institute. Only events with the life cycle attribute value 'complete' are considered and only traces that either have an approved, cancelled or declined application. The event log covers a time period of 6 months and contains around $12,700$ cases and $156,000$ events. The process model of the event log is used to determine the critical moments. A new sub-process in the log is initiated if a customer requests a loan, in that case the Dutch financial institute determines whether an offer will be sent to the customer. In order to determine whether an offer will be sent, human resources are needed to complete the application and to create an offer. If it is known early enough whether an offer will be sent, the resources could be used only for cases in which indeed an offer will be provided to the customer. Therefore, the PoAc is the activity 'O_SENT'. At the activities of 'W_Completeren_aanvraag' (Complete application,W_C_a) and 'A_PREACCEPTED' (A_p) the remainder of the process can still contain the activity 'O_SENT', but the process might also finish without the activity 'O_SENT'. Accordingly, two DeMs are defined, 1) the moment at which 'A_p' occurs and 2) the moment at which 'W_C_a' occurs. For the prediction task on the DeM of 'A_p' only traces in which the activity 'A_p' occurs are considered. Resulting in 6968 traces. For the activity 'W_C_a' the event log also consist of 6968 traces. The occurrence of 'O_SENT' is $67,2\%$ and $44,1\%$ respectively. After creating a training, a validation and a test set for both DeMs, the training set is balanced on the occurrence of 'O_SENT'.

The next step is to convert the traces to inputs for the RFC, LSTM and GAN for both DeMs. For all three models the traces up to the moment 'A_p' as well as the moment 'W_C_a' are used separately as input to train a model for predictions. In the case of the RFC the traces have to be converted to a feature representation. The independent variables contain information of trace and event attributes. The input features of the LSTM network contain similarly information on trace attributes and event attributes.The trace length is set to 3 for 'A_p' and 6 for 'W_C_a' up to the DeM. The preprocessing of [4] is used to create the feature representation. For each trace, additional independent variables are created, which are the amount of loan or overdraft requested by the customer, the number of activities so far, the types of these activities and time between them. The GAN network uses the original prefixes up to the DeM. The input of the training set also contains the suffixes which are either the suffix up to the PoAc or the complete suffix when PoAc is not in the suffix. The feature representation as proposed in [7] is the input for the encoder-decoder GAN.

To determine the best prediction model, the three models and the baseline model should be compared. The result is shown Figure 2(B) and (C). For both DeMs the three models are outperforming the baseline model, as the models score higher on all performance measures. For DeM 'A_p' the F1-score for all three models is comparable. The recall is best on the LSTM model. Therefore, the LSTM model is preferred in predicting 'O_SENT'. For DeM 'W_C_a' the RFC and LSTM model score equally on the F1-score and slightly better than the GAN model. The recall score of the RFC is outperforming those of the LSTM and GAN model. The RFC model is best for the current prediction task.

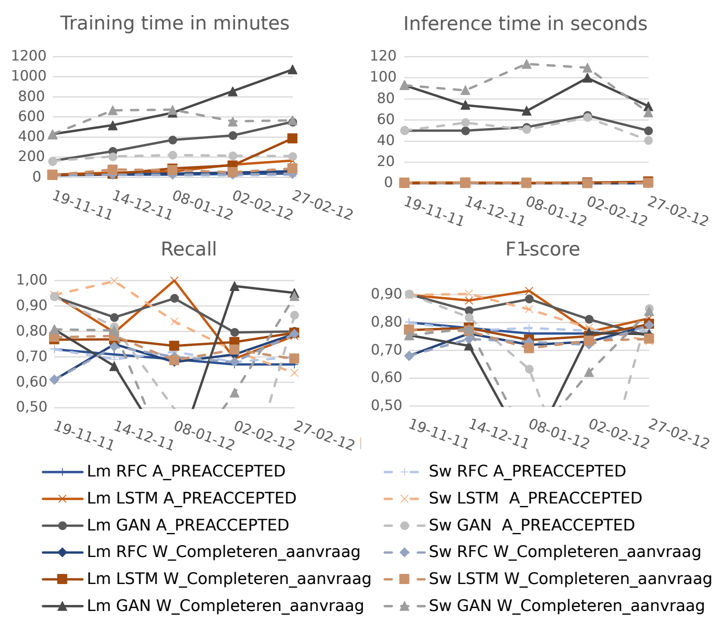

Fig. 4. Sliding window and landmark model results on the BPI 2012 data set.

Next the results of using a sliding window and a landmark model are discussed. The first window consists of 50 days and for each timeframe it either shifts by 25 days (Sw) or 25 days of data are added (Lm). The results for both 'A_p' and 'W_C_a' are shown in Figure 4. For both DeMs it is the case that the training time of the sliding window models is relatively consistent over the different windows, while the training time of the landmark model increases. Resulting in a longer training time for the landmark model for later windows. Furthermore the training time and the inference time of the GAN is longer than the counterparts of the LSTM and RFC model. Considering the performance measure, the GAN model shows some poor performance results for some of the windows. This might be caused by the goal of training for suffix prediction, which is a different goal than predicting a PoAc. For both the RFC and LSTM model the performance (recall and F1-score) for the landmark model are comparable to the sliding window results. Therefore models are not learning form more data and only retraining on the most recent data is needed. For prediction moment 'A_p' the LSTM model is performing better than the RFC model on most windows. Therefore, considering training time, inference time, recall and F1-score the LSTM model with sliding window is preferred. On the other hand, for prediction moment 'W_C_a' the RFC on both windows and the LSTM model on the landmark window perform better than the LSTM with the sliding window. Considering the training time, inference time, recall and F1-score the RFC on the sliding window is preferred.

6 Conclusion

In this paper the HIAP framework is proposed as a repeatable framework for predicting the occurrence of a PoAc at a DeM in the customer journey. Different machine and deep learning models are compared for future predictions of touchpoint of interest using two windowing methods. To show the relevance of the framework, we tested it using two datasets showing the prediction power and the impact of using a sliding window or a landmark window. Showing that the preferred prediction model and windowing technique depends the type of customer journey data. Interesting future research is to predict the moment at which the expected activity is expected to occur [6]. This provides information on the possibility to prevent the activity to occur.

References

1. Bernard, G., Andritsos, P.: Contextual and Behavioral Customer Journey Discovery Using a Genetic Approach. Advances in Databases and Information Systems (2019)
2. van der Veen, G., Ossenbruggen, R.: Mapping Out the Customer's Journey: Customer Search Strategy as a Basis for Channel Management. Journal of Marketing Channelsl **22**, 202–213 (2015)
3. Hassani, M., Habets, S.: Predicting Next Touch Point In A Customer Journey: A Use Case In Telecommunication. In: ECMS 2021 Proceedings **35** (2021)
4. Kratsch, W., Manderscheid, J., Roeglinger, M., Seyfried, J.: Machine Learning in Business Process Monitoring: A Comparison of Deep Learning and Classical Approaches Used for Outcome Prediction. Bus Inf Syst Eng **63**, 261–276 (2021)
5. Taymouri, F., La Rosa, M.:Encoder-Decoder Generative Adversarial Nets for Suffix Generation and Remaining Time Prediction of Business Process Models.arXiv(2020)
6. Tax, N., Verenich, I., La Rosa, M., Dumas, M.: Predictive business process monitoring with LSTM neural networks. In: CAiSE pp.477–492 (2017)
7. Terragni, A.,Hassani, M.: Optimizing customer journey using process mining and sequence-aware recommendation. In: ACM/SIGAPP (SAC), pp. 57–65 (2019).
8. Spenrath, Y., Hassani, M., van Dongen, B.: Online Prediction of Aggregated Retailer Consumer Behaviour. In ICPM Workshops, pp. 211-223 (2021).
9. Aalst, W. M. P. van der.: Process mining: data science in action. Springer (2016).
10. Metzger, A., Neubauer, A., Bohn, P., Pohl, K.: Proactive Process Adaptation Using Deep Learning Ensembles. In: CAiSE, 547–562 (2019).