DEGREE PROJECT IN COMPUTER SCIENCE AND ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2021

Investigating the Attribution Quality of LSTM with Attention and SHAP

Going Beyond Predictive Performance

HANNES KINDBOM

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Investigating the Attribution Quality of LSTM with Attention and SHAP

Going Beyond Predictive Performance

HANNES KINDBOM

Master's Programme, Machine Learning, 120 credits
Date: June 24, 2021
Supervisor at KTH: Hamid Reza Faragardi
Examiner at KTH: Jonas Beskow
School of Electrical Engineering and Computer Science
Host company: Hedvig AB
Supervisor at Hedvig: John Ardelius
Swedish title: En undersökning av attributionskvaliteten av LSTM med attention och SHAP
Swedish subtitle: Bortom prediktiv prestanda

Abstract

Estimating each marketing channel's impact on conversion can help advertisers develop strategies and spend their marketing budgets optimally. This problem is often referred to as attribution modelling, and it is gaining increasing attention in both the industry and academia as access to online tracking data improves.

Focusing on achieving higher predictive performance, the Long Short-Term Memory (LSTM) architecture is currently trending as a data-driven solution to attribution modelling. However, such deep neural networks have been criticised for being difficult to interpret. Interpretability is critical, since channel attributions are generally obtained by studying how a model makes a binary conversion prediction given a sequence of clicks or views of ads in different channels.

Therefore, this degree project studies and compares the quality of LSTM attributions, calculated with SHapley Additive exPlanations (SHAP), attention and fractional scores to three baseline models. The fractional score is the mean difference in a model's predicted conversion probability with and without a channel. Furthermore, a synthetic data generator based on a Poisson process is developed and validated against real data to measure attribution quality as the Mean Absolute Error (MAE) between calculated attributions and the true causal relationships between channel clicks and conversions.

The experimental results demonstrate that the quality of attributions is not unambiguously reflected by the predictive performance of LSTMs. In general, it is not possible to assume a high attribution quality solely based on high predictive performance. For example, all models achieve $\sim 82\%$ accuracy on real data, whereas LSTM Fractional and SHAP produce the lowest attribution quality of 0.0566 and 0.0311 MAE respectively. This can be compared to an improved MAE of 0.0058, which is obtained with a Last-Touch Attribution (LTA) model. The attribution quality also varies significantly depending on which attribution calculation method is used for the LSTM. This suggests that the ongoing quest for improved accuracy may be questioned and that it is not always justified to use an LSTM when aiming for high quality attributions.

Keywords

Digital marketing, Attribution modelling, Multi-touch attribution, Deep learning, LSTM, SHAP, Attention, Interpretability

Sammanfattning

Genom att estimera påverkan varje marknadsföringskanal har på konverteringar, kan annonsörer utveckla strategier och spendera sina marknadsföringsbudgetar optimalt. Det här kallas ofta attributionsmodellering och det får alltmer uppmärksamhet i både näringslivet och akademin när tillgången till spårningsinformation ökar online.

Med fokus på att uppnå högre prediktiv prestanda är Long Short-Term Memory (LSTM) för närvarande en populär datadriven lösning inom attributionsmodellering. Sådana djupa neurala nätverk har dock kritiserats för att vara svårtolkade. Tolkningsbarhet är viktigt, då kanalattributioner generellt fås genom att studera hur en modell gör en binär konverteringsprediktering givet en sekvens av klick eller visningar av annonser i olika kanaler.

Det här examensarbetet studerar och jämför därför kvaliteten av en LSTMs attributioner, beräknade med SHapley Additive exPlanations (SHAP), attention och fractional scores mot tre grundmodeller. Fractional scores beräknas som medelvärdesdifferensen av en modells predikterade konverteringssannolikhet med och utan en viss kanal. Därutöver utvecklas en syntetisk datagenerator baserad på en Poissonprocess, vilken valideras mot verklig data. Generatorn används för att kunna mäta attributionskvalitet som Mean Absolute Error (MAE) mellan beräknade attributioner och de verkliga kausala sambanden mellan kanalklick och konverteringar.

De experimentella resultaten visar att attributionskvaliteten inte entydigt avspeglas av en LSTMs prediktiva prestanda. Det är generellt inte möjligt att anta en hög attributionskvalitet enbart baserat på en hög prediktiv prestanda. Alla modeller uppnår exempelvis $\sim 82\%$ prediktiv träffsäkerhet på verklig data, medan LSTM Fractional och SHAP ger den lägsta attributionskvaliteten på 0.0566 respektive 0.0311 MAE. Det här kan jämföras mot en förbättrad MAE på 0.0058, som erhålls med en Last-touch-modell. Kvaliteten på attributioner varierar också signifikant beroende på vilket metod för attributionsberäkning som används för LSTM. Det här antyder att den pågående strävan efter högre prediktiv träffsäkerhet kan ifrågasättas och att det inte alltid är berättigat att använda en LSTM när attributioner av hög kvalitet eftersträvas.

Nyckelord

Digital marknadsföring, Attributionsmodellering, Multi-touch attribution, Djupinlärning, LSTM, SHAP, Attention, Tolkningsbarhet

Acknowledgments

I would first of all like to express my deep and sincere gratitude towards my close family for their love, caring and sacrifices for educating and preparing me for the future. Special thanks to my partner Rosalind McDermott for her understanding and unconditional support throughout the project.

I would also like to thank John Ardelius, my supervisor at Hedvig, for the project idea and for giving me this opportunity. His visionary mind and encouraging guidance has inspired me. Carl Lager, Sonny Andersson and Emil Wallerstedt from the growth team at Hedvig all deserve recognition as well. I must also thank my supervisor Hamid Reza Faragardi at KTH Royal Institute of Technology for the valuable solution oriented discussions when I have been stuck.

Finally, I cannot express enough thanks to my supportive friend Viktor Reineck. It was an honour to work alongside Viktor during my additional master thesis within mathematics. It would not have been possible to write two reports in parallel without him as the co-author on one of them.

Stockholm, June 2021
Hannes Kindbom

Contents

List of Acronyms

LR Logistic Regression

LSTM Long Short-Term Memory

LTA Last-Touch Attribution

MAE Mean Absolute Error

MTA Multi-Touch Attribution

p.p. percentage points

RNN Recurrent Neural Network

SHAP SHapley Additive exPlanations

SP Simple Probabilistic

TSTR Train on Synthetic, Test on Real

Chapter 1

Introduction

Before becoming customers, users may be exposed to a sequence of online marketing channels. Examples of online marketing channels are email newsletters and Facebook, while TV and billboards are traditional offline channels. Estimating the contribution of each marketing channel during different stages in this, so called, customer journey can help advertisers to develop strategies and spend their marketing budgets optimally. This problem is often referred to as attribution modelling [1], and it is gaining increasing attention in both the industry and academia as access to online tracking data improves. In essence, attribution modelling is defined as the problem of distributing suitable credits to each marketing channel based on its impact on conversion [2], where a conversion refers to a completed sign-up or purchase in this project [3]. A touchpoint along the customer journey could be any type of interaction, such as a click or impression, which adds complexity to the problem.

Attribution modelling may be approached in various ways, either using heuristics or with more advanced data-driven models [4], such as Markov chains [5]. The industry's first commercially available data-driven Multi-Touch Attribution (MTA) model, which considers all touchpoints in a customer journey when estimating empirical probabilities of converting given channel interactions, was proposed in 2011 [6]. Today, the Long Short-Term Memory (LSTM) architecture appears in several papers and is trending as a data-driven solution to achieve higher predictive performance [4, 7, 8].

Yet, such deep neural networks have been criticised for being black boxes, difficult to interpret [7]. Interpretability is critical in attribution modelling, since attributions are usually obtained as each channel's effect on a model's binary conversion prediction. Interpretation has also been pointed out as one

out of three desirable properties of an attribution model, apart from being fair and data-driven [9]. Nevertheless, the procedure of estimating the features' influence, and thus attributions, in LSTMs is non-trivial. Therefore, the suitability of evaluating an LSTM solely on its performance of conversion prediction for attribution modelling may be questioned. Thus, the quality of LSTM attributions in relation to predictive performance will be studied in this degree project. In this context, quality refers to how accurate the calculated attributions are in comparison to ground truth. The attributions' variance will also be studied, certainly as the importance of low variability attribution models has been emphasised [6]. Part of the investigation is also to figure out if LSTMs are beneficial in comparison to baseline models, when the aim is to obtain high quality attributions.

1.1 Objective

The project is done in collaboration with the Swedish insurance company Hedvig. Hedvig's current approach to attribution modelling is rather ad-hoc and characterised by trial and error, yet based on insights from data. Therefore, their overall interest is to find out if their marketing strategy can be automated with data-driven algorithms to optimise for growth. However, due to an aggressive growth strategy, Hedvig expects to encounter new marketing scenarios and increasing amounts of data. Thus, the interest in what model architectures are potentially required for attribution modelling, arises. A hypothesis is that deep learning models like LSTMs could be beneficial when more data containing longer customer journeys are available. From Hedvig's perspective, the objective of this project is therefore to investigate if LSTMs are beneficial on their current dataset or could be useful in a future scenario to obtain attributions of high quality. The quality of the LSTM attributions, as measured by deviation from true attributions, will be benchmarked against three baseline models. These are to be introduced in Chapter 4.

Academically, the objective is to provide a contrasting study to the current focus on predictive performance within attribution modelling. Specifically, insights about the quality of LSTM attributions seem to be unexplored in the existing literature.

1.2 Research Question

In essence, the objective is concretized by the following two research questions:

1. *How is the quality of attributions reflected in the predictive performance of an LSTM?*
2. *Is a deep learning model like LSTM beneficial when the objective is to accurately model attributions?*

1.3 Research Methodology

The research methodology in this project is designed according to Figure 1.1. After identifying Hedvig's needs and a brief look at related work, the research objective is formulated as in Section. 1.1. Following collection and processing of real data, there is an iterative four-step process before conclusions are drawn. However, as mentioned in Section. 1.4, processing of the real data is done within a parallel degree project, wherefore that step is marked with dashed lines in the figure.

A thorough study of related work is the first out of four steps in the iterative process to achieve the research objective. The second step comprises finding a suitable LSTM architecture, a method to calculate attributions as well as metrics and experiments to explicitly validate their quality. A synthetic data generator, further described in Chapter 4, turns out to be an integral part of the attribution validation. The following steps include implementing the selected solution, evaluating it and again study related work if needed.

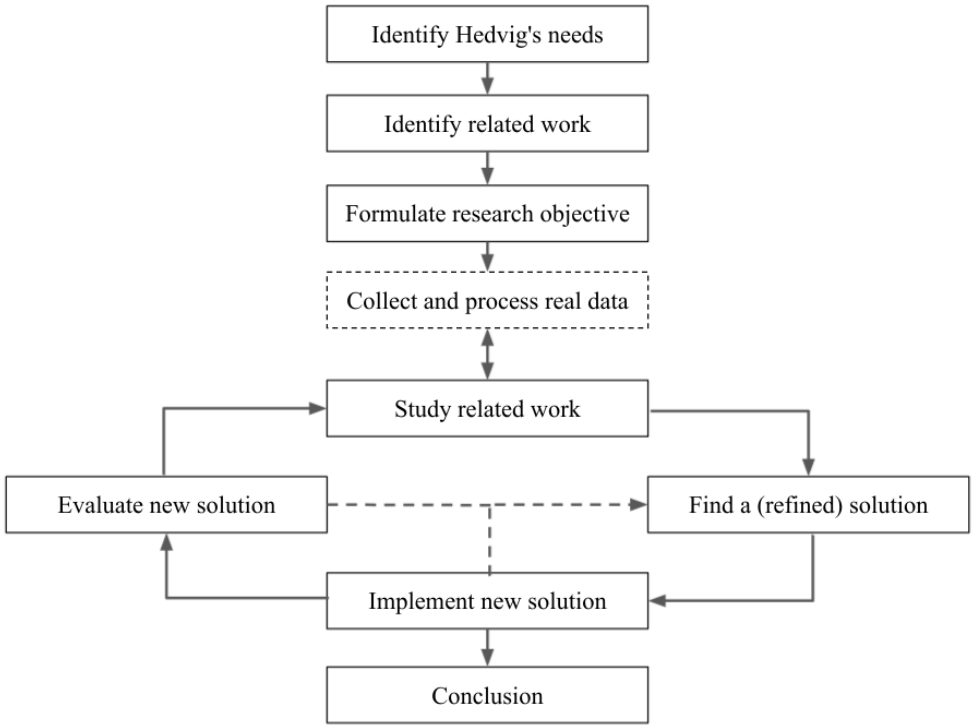

Figure 1.1 – A schematic illustration of the research methodology

1.4 Scope and Delimitations

This project is delimited to using real-world data from Hedvig only. In short, the dataset contains web traffic data collected on the Swedish market during approximately two months. The dataset is restricted to single device tracking and the life span of cookies limits the data quality further. However, processing of this data is considered to be outside the scope and its details and limitations are to be found in another project report, in which the author of this report is also involved [10]. As suggested in the research question, the project is further delimited to developing a single LSTM with a few accompanying attribution calculation methods. The model's predictive performance will be evaluated on its ability to predict whether a customer journey leads to conversion or not, disregarding the monetary value of conversions. Lastly, assessment of when an LSTM could potentially be useful is limited to a synthetic dataset in one hypothetical future scenario of more and longer customer journeys.

1.5 Ethics and Sustainability

Applying data-driven attribution models to optimise marketing efforts has several ethical implications. One of the most considerable matters is the data privacy of users, wherefore it is important to ensure the anonymity of the real website interaction data and comply with laws like General Data Protection Regulation (GDPR) [11]. This project is fully compliant with GDPR to the best of the author's knowledge. For instance, the data were anonymised and no stored data could be connected to individuals.

Regarding the sustainability of the project, it may be argued that the attribution models can be used to more effectively use a marketing company's economic resources, thus improving economic sustainability. However, it is important to emphasise that the attribution models may be used in an environmentally and socially harmful manner if they are applied in unsustainable industries like fast fashion.

Chapter 2

Background

This chapter lays the theoretical foundation for the project and starts with a brief introduction of Recurrent Neural Networks (RNNs) and LSTMs. A description of the attention mechanism and Shapley values is then provided, followed by an overview of the Poisson process.

2.1 RNN

RNN is a type of neural network, which is suitable for processing sequential data, consisting of vectors $\boldsymbol{x_t}$ with time step $t = 1, ..., T$. In contrast to feed-forward neural networks, an RNN has feedback loops. These feedback loops enable memorising information from previous time steps. Particularly, the output $\boldsymbol{h_t}$ of each hidden unit in the forward pass depends on the previous hidden state $\boldsymbol{h_{t-1}}$ and the current input vector $\boldsymbol{x_t}$ as in Equation 2.1 [12].

$$\boldsymbol{h_t} = f(\boldsymbol{h_{t-1}}, \boldsymbol{x_t}) \tag{2.1}$$

Predictions are made based on the outputs $\boldsymbol{h_t}$, usually after applying some activation function like $\tanh$. Activation functions are non-linear functions used to compute the hidden layer values by mapping a real-valued input to a predefined range [12]. The network is then trained using an algorithm called backpropagation through time. Briefly explained, it starts by storing the states in the forward pass and then computing the gradients with respect to the weights in a backward pass [12]. Although RNNs are suitable for learning sequences, it is a challenge to train them on longer sequences [13]. Gradients computed across many time steps tend to either vanish or explode, potentially causing unstable learning or the weights to stop being updated altogether.

2.1.1 LSTM

The LSTM architecture was first developed by Sepp Hochreiter et al. in 1997 to battle the problem of vanishing gradients during backpropagation of vanilla RNNs [14].

Several variations of LSTM exist but all are designed to learn long-term dependencies across sequences. Other pronounced benefits of LSTMs are that they generalise well and are insensitive to choices of hyperparameters, such as learning rate [14]. In essence, each repeating module of the LSTM has four neural network layers that interact. At its core, the cell state carries information that is regulated by three gates. The forget gate layer first decides which information to keep. It is followed by the input gate layer which helps with determining what new information to store in the cell state. The output of the cell state is then filtered and passed on to the next time step. Although mathematical details are omitted in this report, an overview of the LSTM is shown in Figure 2.1 [15].

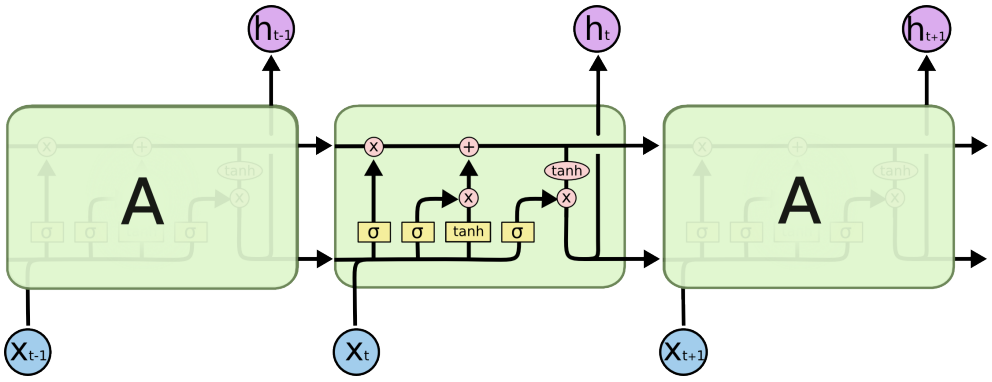

Figure 2.1 – An illustration of how vectors travel through the repeating LSTM architecture. The yellow boxes symbolise learned layers and the pink circles illustrate pointwise operations, such as addition of vectors [15].

2.1.2 Attention

Attention was introduced by Bahdanau et al. in 2015 to improve neural machine translation, which is about creating a single neural network that translates sentences in one language into another. With an attention mechanism, their extended model could automatically search for parts of a sentence that were the most relevant to predict the target word. This improved the performance of the basic encoder-decoder architecture. Briefly speaking,

the traditional encoder network maps a source sentence to a fixed-length vector, and the decoder network maps this vector to a translated target sentence [16]. The more recent Transformer model, which was shown to outperform its predecessors on machine translation tasks, is for example built on attention mechanisms [17].

Since its first introduction, variations of attention mechanisms have been proposed. These can be divided into local and global attentions. The global attention model considers all the hidden states of the encoder, while the local attention focuses only on a small subset of source positions per target. Luong's general global attention is a simplified and generalised version of the original attention, showing improved performance [18]. It is leveraged in this project and can be summarised by the equations below.

$$\alpha_{ts} = \frac{\exp(e_{ts})}{\sum_{s'=1}^{S} \exp(e_{ts'})} \quad \text{where} \quad e_{ts} = h_t^\top W \bar{h}_s \tag{2.2}$$

$$a_t = \tanh(W_c[c_t; h_t]) \quad \text{where} \quad c_t = \sum_{s'=1}^{S} \alpha_{ts'} \bar{h}_{s'} \tag{2.3}$$

W_c and W are trainable weight matrices, and the attention weight α_{ts} can be interpreted as the probability that the target label comes from source (or touchpoint in this report) s, where S is the number of source hidden state vectors [16]. The context vector c_t is then constructed as a weighted sum of all source hidden states, using α_{ts} as weights to capture each touchpoint's importance. h_t is the hidden state of the top layer of the LSTM and $\bar{h}_s$ is the source touchpoint hidden state. After concatenation ($[c_t; h_t]$), the attentional hidden state a_t is then fed through the rest of the model ending up with a prediction [18]. Note that, unlike sequence-to-sequence translation, we only have a single output from our many-to-one LSTM in this project, thus only one target t.

2.2 Shapley Values as Attributions

The Shapley value concept was developed within game theory to determine the importance or value of each player in a multiplayer co-operative game. The Shapley value represents the value of a player across all combinations of player coalitions. In the context of linear regression models, a Shapley value can be assigned to each feature, illustrating its importance for prediction [19]. Each feature's contribution to a coalition is defined as the Shapley value ϕ_i,

where f is the characteristic function that estimates the utility of a coalition. In Shapley regression values, $f_S(\boldsymbol{x_S})$ equals the coefficient of determination R^2 obtained after training a model on features in the subset S. The method involves training a model on all feature subsets $S \subseteq F$, where F refers to the set of all features [20].

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \left[f_{S \cup \{i\}} \left(\boldsymbol{x_{S \cup \{i\}}} \right) - f_S \left(\boldsymbol{x_S} \right) \right] \tag{2.4}$$

As opposed to Shapley regression values, there is no need to retrain the model on all feature subsets in the Shapley sampling values method. Instead, sampling approximations are applied to Equation 2.4 and the effect of removing a feature from the model is approximated by integrating over training samples [20]. However, the details of Shapley sampling values are outside the scope of this report.

2.3 Poisson Process

The history of using the Poisson process to model website traffic goes back to year 2003, where it was used as part of a Markov Modulated Poisson Process (MMPP). An MMPP is a type of Poisson process whose arrival rate varies according to a Markov process [21].

Before constructing more sophisticated models for website traffic, a description of the standard Poisson process should be given. If $X(t)$ is the number of event occurrences by time $t \geq 0$ and $P(X(t) = n)$ is defined as in Equation 2.5, the counting process $\{X(t); t \geq 0\}$ is called a Poisson process with constant mean rate λ [22].

$$P(X(t) = n) = \frac{(\lambda t)^n e^{-\lambda t}}{n!} \quad n = 0, 1, 2... \tag{2.5}$$

Furthermore, the random variable $X(t)$ follows the Poisson distribution with expected value $E[X(t)] = \lambda t$. The Poisson process is said to be memoryless, meaning that every time interval of fixed length has the same probability of containing an event, regardless of when the preceding event occurred [22]. In other words, the occurrences of events in disjoint time intervals are assumed to be independent.

Another property of the Poisson distribution is that the time between consecutive events follows an exponential distribution with rate parameter

λ. The probability density function of such a memoryless exponentially distributed variable T is defined as in Equation 2.6 [22]. It is commonly used to model times between arriving customers within queueing theory.

$$f_T(t) = \begin{cases} \lambda e^{-\lambda t}, & \text{for } t \geq 0 \\ 0, & \text{for } t < 0 \end{cases} \tag{2.6}$$

Chapter 3

Related Work

This chapter is dedicated to a thorough review of academic research related to attribution modelling. After a gentle introduction to the broader field of attribution modelling, the focus will be on various deep learning approaches, since this degree project is focused on such techniques.

3.1 Attribution Modelling

Attribution modelling has been approached with both rule-based methods and more sophisticated data-driven algorithms [7]. Last-Touch Attribution (LTA) is a common rule-based method, which assumes that the conversions are solely caused by the last advert that was clicked or viewed before conversion [7]. An example of a traditional data-driven attribution model is the discrete-time Markov chain, where each advertisement click is modeled as a state [5].

In general, the problem is approached in a supervised manner and the models are evaluated based on their performance of conversion prediction, either measuring the Area Under the receiver operating characteristic Curve (AUC) or accuracy [8]. Common for all approaches is that their ultimate purpose is to optimise for some form of business metric, such as Return On Investment (ROI).

3.2 Data and Features

Regarding data, some models have been trained on aggregated data across customer journeys. For example, during a study with higher-order Markov chains, Google Analytics data on the number of conversions for each unique

customer journey were used, although the author expressed a desire to use more detailed data [5]. However, most of the time, the models are trained on user-level data, where the Criteo dataset is a publicly available example.*

Depending on the nature of the applied model, the feature engineering process differs. For logistic regression, the features may for instance be the number of clicks via each channel in a customer journey [23]. For some deep learning models, user-context information, such as age, has been incorporated in the model together with a one-hot representation of the touchpoints [8].

3.3 Model Architecture

During a comprehensive literature review on the topic, application of Bayesian, probit-, logit-, and tobit regression, Markov chains, game theory, ensemble- as well as deep learning models were found within the field [24]. Current state-of-the-art within deep learning is to use some variant of LSTM. One recent study proposed a phased-LSTM combined with Shapley values and linear regression to increase interpretability [7]. In another paper, an LSTM with attention including user features was proposed [8]. An LSTM with attention to both ad-impressions and clicks has also been proposed together with a framework for back-evaluating the profitability of using the attributions in practice [4].

Although deep learning models have been found to outperform simpler models like logistic regression in terms of predictive performance, it is still a relatively unexplored model type for this application. As recently as year 2018, the authors themselves claimed that they have built the first deep learning algorithm for a MTA problem in marketing [24].

3.4 LSTM Attributions

The actual attributions are, in general, obtained as each channel's influence on prediction after the classifier has been trained on the task of predicting whether a customer journey will result in conversion or not. It is well known that LSTMs are generally difficult to interpret [7] and their feature importance can be estimated in different ways. However, several techniques to overcome this disadvantage have been suggested.

The authors of [4] used a dual-attention mechanism to calculate attribution values via the conversion prediction. They constructed a

* http://apex.sjtu.edu.cn/datasets/13

sequence-to-sequence LSTM to model both clicks and impressions of ads. The attention was applied to both the touchpoint features as well as over the hidden states of clicks. This dual-attention was said to contribute to prediction accuracy while naturally forming the attributions. Specifically, attributions for each touchpoint were obtained as a weighted sum of the click- and impression attention weights. Attributions per channel were then obtained by aggregating the touchpoint attributions on channel level for conversions only. Single attention has also been applied together with an LSTM [8] to obtain attributions in a similar manner. In the same study, attributions were also calculated as fractional scores, i.e., as the aggregated difference in conversion probabilities with and without the channel.

As mentioned above, in an attempt to make the attribution values more interpretable, a framework for calculating attributions using a phased-LSTM, Shapley values and linear regression has been published [7]. The attribution of each touchpoint was calculated from the coefficients of a linear regression model built on top of Shapley values. Shapley regression values, described in Section 2.2, were used for shorter customer journeys and Shapley sampled values for longer journeys.

Outside of the attribution modelling field, several well-established techniques to calculate the influence of each feature exist. In 2017, Scott M. Lundberg and Su-In Lee from the University of Washington developed SHapley Additive exPlanations (SHAP), a framework for interpreting predictions of any machine learning model. Compared to earlier methods, it demonstrated improved computational efficiency and alignment with human intuition in experiments. In essence, the SHAP value for a feature is the difference in expected model prediction when conditioning on that feature. Although the exact SHAP values are difficult to compute, they can be approximated in different ways [20]. The Integrated Gradients is another method to explain the output of a prediction model. The principle of Integrated Gradients is to calculate the path integral of the neural network gradients along the straight line path between the baseline input and the input at hand. The choice of baseline input depends on application but it could be a zero embedding vector [25].

Scott M. Lundberg later published a code repository containing the GradientExplainer*. It is an implementation of SHAP in combination with the Integrated Gradients algorithm and SmoothGrad [26]. In GradientExplainer, the entire dataset is used as the background distribution instead of a single reference value. Specifically, expected gradients are used to calculate

* https://github.com/slundberg/shap

approximate SHAP values for deep learning models under the assumption that input features are independent.

3.5 Research Contribution

In summary, LSTMs have gained attention recently while the quality of their attributions seems to be unexplored. As described in Chapter 1, this degree project is therefore focused on explicitly studying the quality of an LSTM's attributions rather than merely evaluating its predictive performance.

Due to the promising results of SHAP and Integrated Gradients in other application domains, interest arises as to whether GradientExplainer can be used for attribution modelling. In conclusion, it is adopted in this degree project as one of the methods for attribution calculation. To compare to current research in deep learning attribution modelling, GradientExplainer is benchmarked against the above-mentioned attention and fractional scores.

Chapter 4

Method

In this chapter, a detailed description of the synthetic data generation is given, following a brief overview of the Hedvig data. Thereafter, tuning of the model's hyperparameters and implementation of attribution calculation methods are described. Finally, experiments are described in detail. The code is available through a Github repository *.

4.1 Data

In order to test the model in a real-world setting as well as to accurately validate the attributions, both real and synthetic data were used. Collection and processing of the real Hedvig data is part of the parallel degree project [10], whereas a synthetic data generator was developed as part of this project. Therefore, only a summary of the real data processing is given in this section.

4.1.1 Real Data

Historical data from Google Analytics on user touchpoints over marketing channels were supplied by Hedvig. It was collected between 2021-02-03 (denoted t_0) and 2021-04-07 (denoted T). The raw data were organised by user website interaction and each interaction contained information about client id, session id, timestamp, channel, device type and whether it resulted in a conversion or not. Henceforth, sessions are referred to as clicks and a conversion means a user starting an insurance subscription. Only customer journeys with the first click within a subinterval $(t_A = \text{2021-02-24}, t_B = \text{2021-03-17})$ were extracted from the raw data to form a cohort, subject

* https://github.com/hkindbom/attribution-modelling

to analysis. Extracting a cohort with $t_0 < t_A < t_B < T$ in this way reduces the number of false negatives in the dataset, although some users may still convert after time T. Additionally, downsampling was performed by randomly extracting a subset of the majority class to counter class imbalance. After further processing, sequences (or customer journeys) $\boldsymbol{c_i}$ of time ordered channel clicks, 811 of each class, with corresponding binary labels $y_i \in \{0, 1\}$ were obtained for $i = 1, ..., n$. $y_i = 1$ denotes a positive customer journey resulting in conversion and vice versa. An overview of two data samples post-processing is shown in Table 4.1. Further details of the real data can be found in [10].

Table 4.1 – Example rows from the processed dataset [10]

Client	Session	Timestamp	Channel	Device	Conversion
10435	16142	2021-02-25 10:10:38	TikTok	Mobile	0
28588	29441	2021-03-03 17:02:15	Facebook	Desktop	0
	32817	2021-03-11 08:33:54	Google Paid	Desktop	0
	33902	2021-03-12 10:40:32	Google Organic	Desktop	1

4.1.2 Synthetic Data

The motivation to create synthetic data is multifaceted. Firstly, it can be used to easily study different marketing strategies, which makes the results more general. However, most importantly, it arguably enables accurate validation of a model's channel attributions, since the true causal relationships between channel clicks and conversions are thus known and under control. For example, an attribution model should assign more credits to a channel that is simulated to have a bigger influence on conversion. Since the type of marketing data used for attribution modelling is seldom publicly available, such a data generator also contributes to data standardisation within the field and makes the experiments reproducible.

Synthetic data have been generated for similar purposes before. In one study, data were generated according to a multivariate normal distribution by controlling channel exposure correlations, likelihood to show an ad to a user and the probability of converting conditioned on seeing an ad [9]. A benefit of this approach is that control of correlations between channel views enables simulation of different target groups. For instance, it is a reasonable assumption that some users exist only on certain channels. On the contrary, a user cannot be exposed to an ad more than once per channel in their model. Another arguably unrealistic property is that the perspective of continuous

time is missing in their approach. This generator was not validated against real data, nor made publicly available, which further motivates development of a new data generator.

As mentioned in Section 2.3, the Poisson process has been used to model website traffic before. This stochastic process is commonly used to model events occurring over time [27]. Below follows a description of the generator that was developed in this project, built on Poisson processes and inspired by the above mentioned generator [9].

Before simulation, a cohort of N persons is created alongside K channels. Each person has a base probability p_c of clicking on an ad when exposed to it. A person's base probability of converting following a click is p_b. Multiple unique properties are also assigned to every channel. p_{kc} refers to the increase in base click probability a person experiences after clicking through channel k. Similarly, the increase in base conversion probability is denoted p_{kb} for channel k. In other words, both p_c and p_b are incremented every time a person clicks on a channel. The rate at which a channel k is shown independently to every user is denoted by λ_k.

When simulation starts, each channel k is shown to each user according to a Poisson process. The times between consecutive ad exposures are then independent and exponentially distributed with rate parameter λ_k. When viewing an ad, a person clicks on it according to Equation 4.1, where f_c is a constant momentary click factor and p_{max} is the maximum probability of clicking or converting.

$$P(\text{click} \mid \text{view } k) = \min(p_{max}, p_c + f_c \cdot p_{kc}) \tag{4.1}$$

The succeeding probability of converting after clicking on channel k is illustrated in Equation 4.2, where f_b refers to the constant momentary conversion factor. These momentary factors f_c and f_b may be used to assign a weight to a person's spontaneity in relation to their memory of the brand.

$$P(\text{convert} \mid \text{click } k) = \min(p_{max}, p_b + f_b \cdot p_{kb}) \tag{4.2}$$

The simulation runs until the time limit T is reached or until all N persons have converted.

4.1.3 Synthetic Data Validation

Train on Synthetic, Test on Real (TSTR) was introduced in a paper from 2017 as a metric to evaluate the synthetic output data quality of Generative

Adversarial Networks (GANs) [28]. GANs are popular when it comes to generating realistic-looking data. The authors claim that TSTR shows the ability to use synthetic data for real applications, which is aligned with the objectives of this project. However, TSTR has also been used in a non-GAN exclusive context. The mean TSTR for a synthetic data generation method S across multiple predictive models can be seen in Equation 4.3 [29]. M refers to the number of predictive models and $\mathcal{A}_i(\mathcal{D})$ denotes model i trained on dataset $\mathcal{D}$. Furthermore, $\mathcal{D}^S$ is the synthetic training set and $\mathcal{D}^R$ is the real dataset used for testing. m is a performance metric taking a trained model and a test set as inputs and returns a real value.

$$\mathrm{TSTR}(S) = \frac{1}{M}\sum_{i=1}^{M} m\left(\mathcal{A}_i\left(\mathcal{D}^S\right), \mathcal{D}^R\right) \tag{4.3}$$

In the context of this degree project, TSTR was applied to a test set $\mathcal{D}^R$ of equally balanced classes, corresponding to 20% of the real dataset. Clearly, the class balance was set to be the same for both real and synthetic data. If the generation created excessive data of any class, it was downsampled. Hyperparameters, such as N and f_c, were manually fine-tuned until $\mathrm{TSTR}(S)$ was sufficiently close to the model performance when training on the same amount of real data, equivalent to the remaining 80%. The rate parameter λ_k for each channel k was selected based on the occurrences of clicks per channel in real data [10].

To supplement the TSTR validation metric, the customer journey lengths of the synthetic data were compared to the real data. Altogether, these two methods were considered to be a sufficient amount of validation for the scope of this project.

4.2 Model Implementation

The implementation details of the LSTM alongside the methods for attribution calculation are described in this section. The LTA, Logistic Regression (LR) and Simple Probabilistic (SP) models were compared to in the experiments. However, these baseline models are described in the parallel report [10].

4.2.1 LSTM

The single layer LSTM was implemented using the Sequential model in Keras with Tensorflow as backend. The commonly used Adam optimiser was used

with learning rate $\eta = 0.001$ for training. This default value is similar to what was used in one of the related works and was therefore deemed to be adequate [7]. Adam optimisation is a memory efficient type of stochastic gradient descent, where first- and second-order moments are estimated [30]. It was leveraged in one of the related works, which further increases its relevance [8]. Since it is well suited for binary classification problems, binary cross-entropy was used like in [4] as loss function with sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$ as activation function in the output layer. Sigmoid is appropriate to use for binary classification, as it ensures that the output $\hat{y}$ is on the interval $(0, 1)$, corresponding to the predicted probability of label 1. After some experimentation on the real dataset, 32 output units were found to yield high predictive performance on the 20% validation set. Various combinations of batch size and number of epochs were tried before finding a satisfactory setting of batch size of 20 and 10 epochs. For hyperparameters not mentioned, default values were used. No regularisation was deemed to be necessary as the training- and validation accuracy remained similar throughout training.

Regarding input data and feature engineering, each click sequence $\boldsymbol{c_i}$ was transformed to a sequence of one-hot encoded vectors $\boldsymbol{x_t} \in \mathbb{R}^K$, where K is the number of channels. Tensorflow's Masking was then used to account for variable length input sequences.

4.2.2 Attention Attributions

After constructing the basic LSTM, an attention mechanism was added to it as per Section 2.1.2. Attention has been claimed to improve predictive performance while simultaneously being a natural method to learn attributions over the click sequence [4]. For each batch, the attention outputs a matrix of size $(\text{batch size}, 128)$. Following training, attention weights α_{is} for each touchpoint s in each input sequence i then formed the basis for attribution calculation. Specifically, the unnormalised attribution Attr'_k for channel k is given in Equation 4.4 as a mean attention weight across all sequences predicted as conversion.

$$\text{Attr}'_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \sum_{j=1}^{m_i} \lfloor \hat{y}_i \rceil \alpha_{ij} \mathbb{1}(c_{ij} = k) \tag{4.4}$$

where $\mathbb{1}(\cdot)$ is the indicator function, n_k the number of sequences containing channel k, and m_i is the length of sequence i. To enable comparison between attribution methods, all attributions were finally normalised, i.e.,

$\text{Attr}_k = \frac{\text{Attr}'_k}{\sum_{\forall l} \text{Attr}'_l}$.

4.2.3 SHAP Attributions

The GradientExplainer was also used as an alternative method of attribution calculation for the LSTM. As explained in Section 3.4, GradientExplainer approximate SHAP values, where a channel increases the probability of conversion if its corresponding SHAP value is positive and vice versa for a negative SHAP value. As for the attention attributions, a mean SHAP value across all sequences, now including non-conversions, was used as unnormalised attribution Attr'_k for each channel k. Note that negative attributions were adjusted to 0.

4.2.4 Fractional Attributions

The fractional score was also used as a less advanced alternative to attribution calculation. As described in Section 3.4, it has previously been used to obtain attributions from a trained LSTM, which motivates its relevance. The unnormalised attribution for a channel k is estimated as the mean difference in conversion probability with and without that channel over all customer journeys [8]. A mathematical formula for the fractional score is shown in Equation 4.5, where $\hat{y_{ik}}$ is the predicted probability of conversion for sequence i when all clicks on channel k are removed from it. Like in Section 4.2.2, n_k denotes the number of sequences containing at least one click on channel k.

$$\text{Attr}'_k = \max\left(\frac{1}{n_k}\sum_{i=1}^{n_k}(\hat{y_i} - \hat{y_{ik}}), 0\right) \tag{4.5}$$

4.2.5 Touchpoint Importance

For improved interpretability of the LSTM's predictions, attributions were also calculated on a touchpoint level as in Equation 4.6 on the real dataset.

$$\text{Attr}'_s = \frac{1}{n_m}\sum_{i=1}^{n_m} y_i \alpha_{is} \quad \text{, for} \quad s = 1, ..., m \tag{4.6}$$

where y_i is the binary label for each sample i and n_m is the number of sequences of length m. Every Attr'_s was computed and subsequently normalised for all fixed length sequences $m \in \{2, 3, 4, 5\}$ separately.

4.3 Experiments

After the model and its corresponding attribution calculation methods were implemented and optimised, several experiments could be set up to investigate the research questions. The details of these experiments are described in this section.

4.3.1 Predictive Performance Evaluation

The models' predictive performance was measured both on real and synthetic data after training in all experiments. Some of the more common metrics to evaluate binary classifiers' predictions are listed as in Equation 4.7, where $n = TP + TN + FP + FN$ denotes the total number of samples in the dataset. Here, TP = True Positives, TN = True Negatives, FP = False Positives and FN = False Negatives.

As opposed to precision and recall, accuracy has been criticised for being misleading when used on unbalanced datasets [31, 32]. As described in Section. 4.1.1, classes were balanced and accuracy was therefore still deemed to be a suitable primary metric for evaluation. However, just like in the parallel project, all four metrics were reported for transparency [10].

$$\text{Accuracy} = \frac{TP + TN}{n} \tag{4.7a}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4.7b}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4.7c}$$

$$\text{F}_1 = \frac{TP}{TP + \frac{1}{2}(FP + FN)} = 2\,\frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.7d}$$

4.3.2 Validation of Attributions

The true causal relationship between channel clicks and labels is unknown in the real dataset, wherefore calculated attributions cannot be compared to any ground truth in that case. As mentioned in Section 4.1.2, the synthetic data generator may be used to circumvent this issue. In particular, the conversion increase probability p_{kb} forms the true attribution for channel k. Thus, the Mean Absolute Error (MAE) between normalised values $\bar{p_{kb}}$ and Attr_k was used as a heuristic metric for a model's attribution quality. The exact formula

is shown in Equation 4.8. To capture the variability, it was supplemented with the mean attribution standard deviation across channels, where a channel's attribution standard deviation was obtained with 5-fold cross-validation. The MAE was first computed on the synthetic data after it had been fine-tuned to replicate the real data. Then, it was computed in the hypothetical scenario, described further in Section 4.3.3.

$$\text{MAE} = \frac{1}{K}\sum_{\forall k} |\bar{p_{kb}} - \text{Attr}_k| \quad \text{, where} \quad \bar{p_{kb}} = \frac{p_{kb}}{\sum_{\forall j} p_{jb}} \tag{4.8}$$

4.3.3 Hypothetical Scenario

Some hyperparameters of the data generator were then modified to make the same evaluation in a hypothetical, yet realistic, scenario. The motivation behind the creation of this fictive scenario was to study if the advanced LSTM architecture would be more useful with more data and longer sequences, in comparison to the baseline models LTA, LR and SP. The exact modifications of hyperparameters, compared to the scenario where real data were replicated, are shown in Table 4.2. A larger N was selected in order to increase the amount of data, while smaller λ_k's and f_b in combination with a larger T were intended to make the sequences longer. Intuitively, the smaller f_b means that users were assumed to be less spontaneous and rather affected by previous clicks when making purchase decisions.

Table 4.2 – Modification of hyperparameters in the hypothetical scenario

Hyperparameter	Real	Hypothetical
T	22	100
N	12000	40000
f_b	9	0.5
λ_k	-	real $\cdot \frac{4}{9}$

4.3.4 Tools Used

An overview of the tools used in this project, along with a short description of each tool's purpose is stated in Table 4.3.

Table 4.3 – A list of programming tools used in the project

Tool	**Version**	**Purpose**
Python	3.7	Programming language
Numpy	1.19.5	Matrices and mathematical operations
Tensorflow	2.4.1	Backend for LSTM implementation
Keras	via Tensorflow	High level framework for LSTM implementation
attention	3.0	Attention Attributions
heapq	via Python	Priority queue for synthetic data generation
shap	0.39.0	SHAP Attributions using GradientExplainer

Chapter 5

Results

The results from the experiments, described in Chapter 4, are presented here. The models' predictive performance along with obtained attributions are presented in both the real and hypothetical scenario, following figures and tables from the synthetic data validation. The chapter ends with a presentation of the results from the attribution validation.

5.1 Synthetic Data Validation

After fitting the synthetic data to the real balanced data, the general hyperparameters in Table 5.1 were shown to yield the highest mean TSTR and the most similar customer journey length distribution in Figure 5.1. The time limit T was set to the same length as the real cohort time interval and N large enough to yield enough sequences of each class.

Table 5.1 – The general hyperparameters that made the synthetic data the most similar to the real data as defined by the specified metrics TSTR and customer journey length distribution

Hyperparameter	Value
T	22
N	12000
K	10
p_c	0.02
p_b	0.02
f_c	1
f_b	9
p_{max}	0.85

In addition to these general hyperparameters, Table 5.2 shows the channel-specific hyperparameters that were used to validate the generator.

Table 5.2 – The channel-specific hyperparameters that made the synthetic data the most similar to the real data as defined by the specified metrics TSTR and customer journey length distribution

Channel	p_{kc}	p_{kb}	λ_k
Direct	0.055	0.055	0.02813
Adtraction	0.065	0.065	0.01688
Facebook	0.005	0.005	0.05625
Google Paid	0.055	0.055	0.03656
Google Organic	0.050	0.050	0.03938
LinkedIn	0	0	0.01406
Newsletter	0.075	0.075	0.01406
Snapchat	0	0	0.01688
Studentkortet	0.065	0.065	0.01688
Tiktok	0	0	0.04219

The mean TSTR, together with its value per model, is displayed in Table 5.3. As may be noted, the TSTR values are almost identical to when training and testing on real data only, irrespective of model and performance metric.

Table 5.3 – The TSTR for all models and the corresponding mean TSTR using four different prediction metrics in %

Model	Training set	Accuracy	Precision	Recall	F_1
Mean of all	Real	81.79	75.07	95.35	84.00
	Synthetic	81.87	75.10	95.51	84.09
LSTM	Real	81.48	74.09	93.46	82.66
	Synthetic	81.79	74.23	94.12	83.00
SP	Real	81.48	74.88	95.76	84.04
	Synthetic	81.48	74.88	95.76	84.04
LR	Real	82.72	76.42	96.43	85.26
	Synthetic	82.72	76.42	96.43	85.26
LTA	Real	81.48	74.88	95.76	84.04
	Synthetic	81.48	74.88	95.76	84.04

As a supplement to validation with the TSTR metric, the customer journey lengths were also shown to coincide. The customer journey length distribution of the real- and synthetic data is illustrated in Figure 5.1.

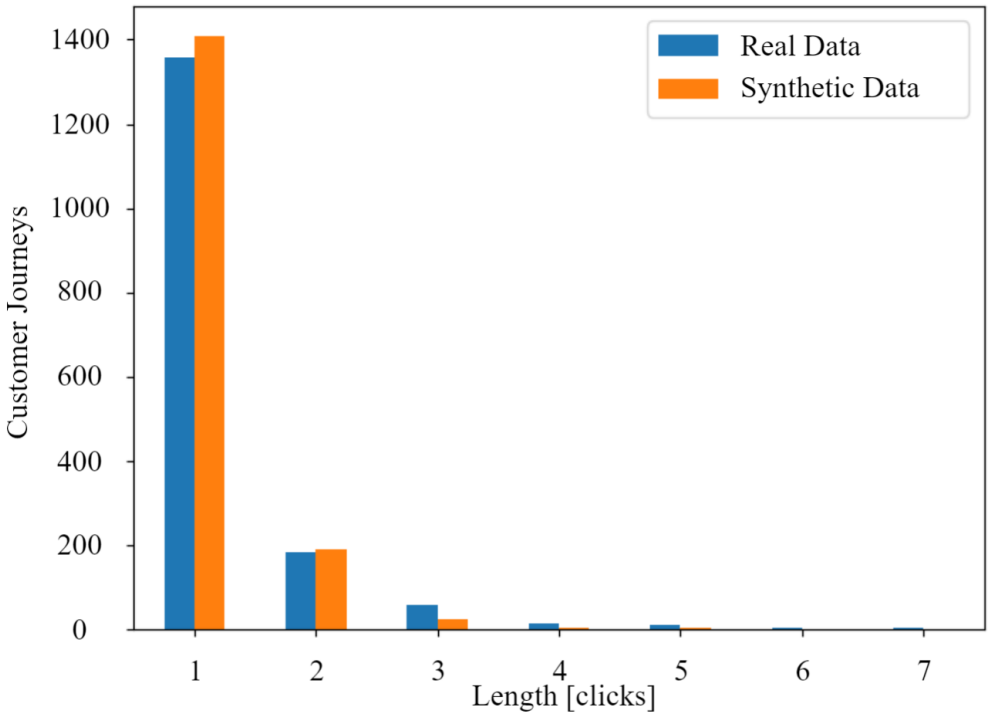

Figure 5.1 – Validation of synthetic data by the length of all customer journeys after balancing of classes.

5.2 Real Scenario

5.2.1 Predictive Performance

The predictive performance of LSTM compared to the baseline models on real data is illustrated in Table 5.4. Specifically, the accuracy may be compared to the theoretical maximum accuracy of 82.31%, calculated for all data samples by predicting each unique sequence as its most common label. The theoretical maximum accuracy and predictive performance of LR, LTA and SP were imported from the parallel degree project [10].

Table 5.4 – Predictive performance of models trained and evaluated with 5-fold cross-validation on the entire real dataset with class balance

Model	Accuracy	Precision	Recall	F_1
LSTM	82.16%	75.25%	96.07%	84.39%
SP	82.16%	75.10%	96.16%	84.33%
LR	82.04%	75.09%	95.91%	84.23%
LTA	82.16%	75.10%	96.16%	84.33%

5.2.2 Touchpoint Importance

Figure 5.2 displays the importance of the order of touchpoints obtained with the attention mechanism on real data. Due to lack of data on longer click sequences, only the normalised mean attention values for positive customer journeys of fixed lengths ranging from 2 to 5 were studied. Observe that the importance of clicks increases closer to conversion for all sequence lengths.

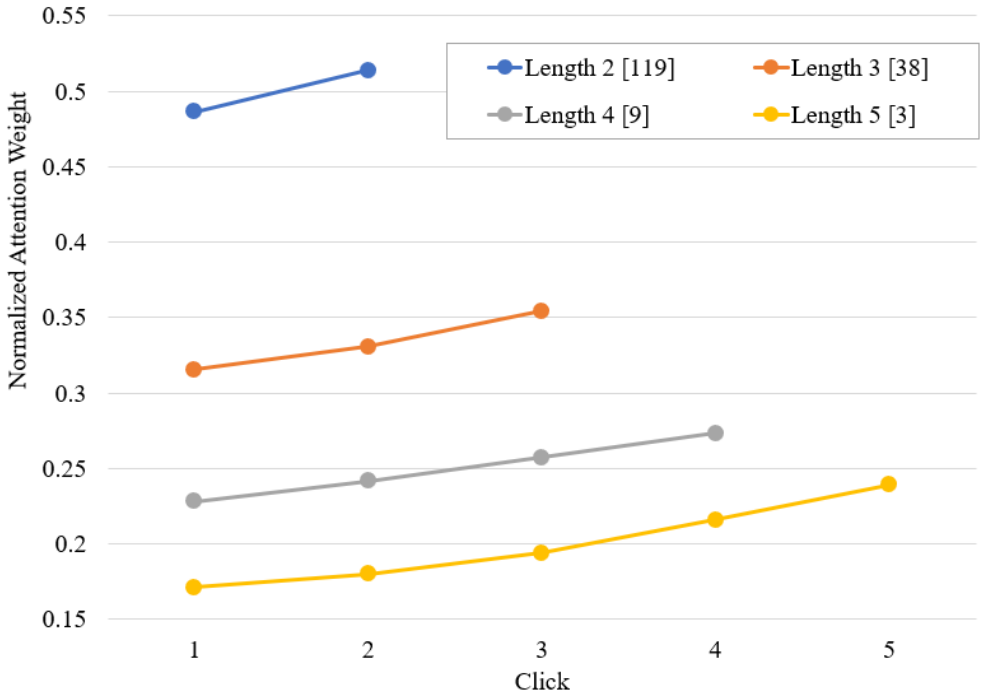

Figure 5.2 – Normalised touchpoint importance obtained with attention for fixed lengths positive customer journeys on balanced data. The numbers in brackets display how many sequences of a given length.

5.2.3 Obtained Attributions

Mean attributions were calculated on the real dataset with 5-fold cross-validation and balanced classes. To illustrate the stability of each attribution model, the standard deviation is displayed as an error bar along with the attributions in Figure 5.3. Table 5.5 shows the mean standard deviation for each model across channels as a summarising metric of the stability. It should be pointed out that LTA and LSTM with attention display the lowest variance, while LSTM with fractional attributions yields the highest variability.

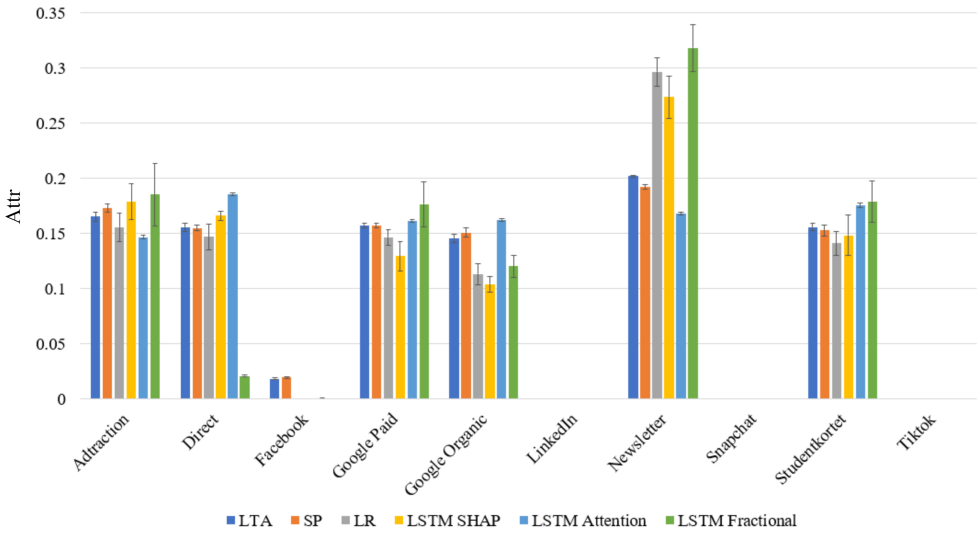

Figure 5.3 – The mean attributions and corresponding standard deviation as error bars for the real dataset obtained with 5-fold cross-validation

Table 5.5 – Attribution mean standard deviation per model for balanced real data

Model	Mean std
LSTM SHAP	0.00791
LSTM Attention	0.00087
LSTM Fractional	0.00999
SP	0.00206
LR	0.00652
LTA	0.00181

5.3 Hypothetical Scenario

Following the modification of general hyperparameters in Table 4.2, the customer journey lengths were shown to increase. Figure 5.4 illustrates this phenomenon. For fair comparison, the synthetic data were downsampled in this figure to the same size as the real data.

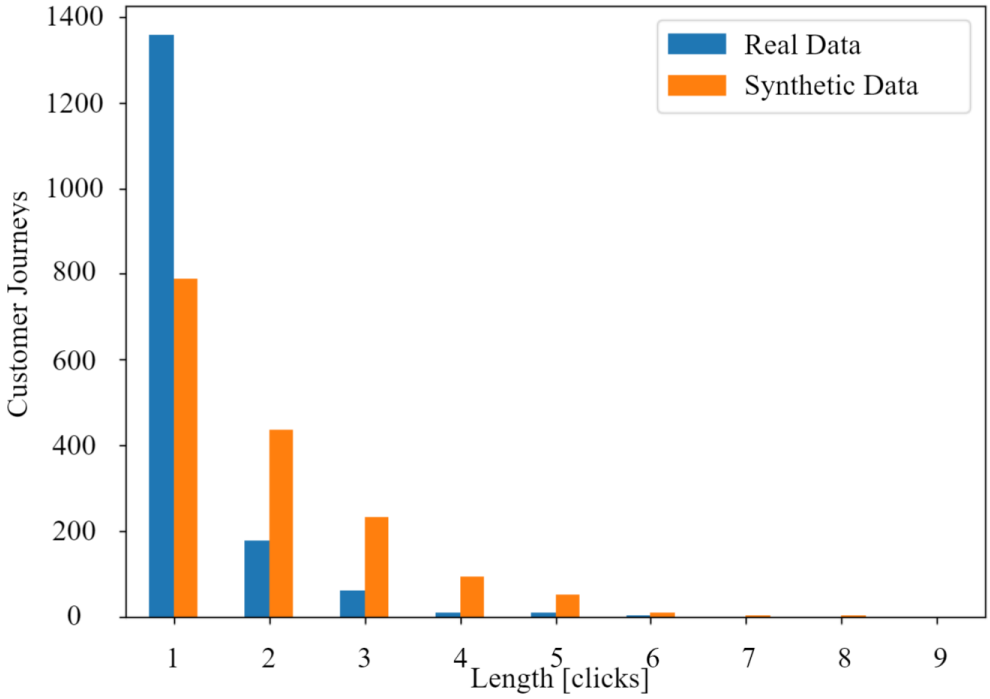

Figure 5.4 – Customer journey lengths in the hypothetical scenario compared to real data with balanced classes of the same size in both datasets

5.3.1 Predictive Performance

After balancing of classes, the hypothetical scenario resulted in 2500 positive and negative customer journeys respectively. This is more than three times the data in the real scenario. It may be observed that LSTM achieves the highest accuracy and that the relative performance of LTA and SP dropped compared to on real data. Table 5.6 shows the predictive performance in this scenario where the theoretical maximum accuracy was 70.34%.

Table 5.6 – Predictive performance of models trained and evaluated with 5-fold cross-validation in the hypothetical scenario with class balance

Model	Accuracy	Precision	Recall	F_1
LSTM	64.48%	64.84%	63.41%	64.11%
SP	55.30%	52.89%	97.64%	68.61%
LR	62.66%	67.60%	48.48%	56.46%
LTA	51.22%	51.19%	56.76%	53.83%

5.3.2 Obtained Attributions

Figure 5.5 shows the calculated attributions in the hypothetical scenario along with the true attributions. It is apparent that the attributions from LTA, SP and LSTM with attention are approximately equally distributed across channels. For instance, they all attribute positive values to LinkedIn, Snapchat and Tiktok, although these channels have 0 in true attribution. In contrast, LR, LSTM SHAP and LSTM Fractional vary more, in accordance with the true attributions.

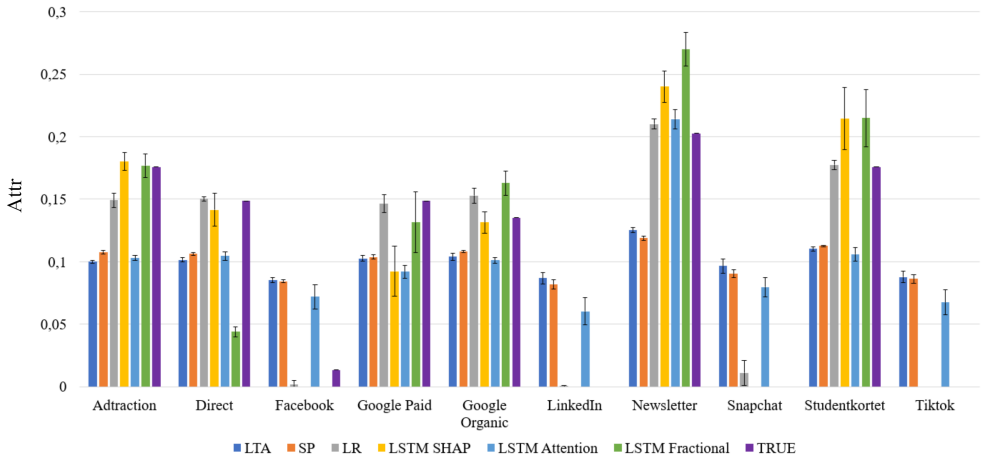

Figure 5.5 – The mean attributions and corresponding standard deviation as error bars in the hypothetical scenario obtained with 5-fold cross-validation

5.4 Validation of Attributions

Table 5.7 shows the results after validation of real data attributions as well as validation of attributions in the hypothetical scenario, both using synthetic data. Apart from that, the same setup as before was used for both scenarios.

As suggested in Section 5.3.2, LR, LSTM SHAP and LSTM Fractional are more accurate in terms of attributions in the hypothetical scenario. Thus, they achieve a lower MAE.

Table 5.7 – Validation of real data attributions as well as validation of attributions in the hypothetical scenario, both using synthetic replicas and 5-fold cross-validation. The lowest values in each column and for each scenario are marked in bold

Model	Scenario	MAE	Mean std
LSTM SHAP	Real	0.0311	0.0080
	Hypothetical	0.0161	0.0086
LSTM Attention	Real	0.0130	**0.0012**
	Hypothetical	0.0554	0.0064
LSTM Fractional	Real	0.0566	0.0117
	Hypothetical	0.0271	0.0083
SP	Real	0.0191	0.0026
	Hypothetical	0.0659	**0.0019**
LR	Real	0.0193	0.0054
	Hypothetical	**0.0080**	0.0041
LTA	Real	**0.0058**	0.0027
	Hypothetical	0.0686	0.0029

Chapter 6

Discussion

This chapter starts with an interpretation of the results, followed by a validity discussion, conclusions and finally suggestions for future work.

6.1 Interpretation of Results

6.1.1 Synthetic Data Validation

Table 5.3 shows that the mean TSTR accuracy across models is $\sim 81.9\%$. This number is high, considering that it exceeds the mean accuracy when training and testing on real data only. As argued in Section 4.1.3, this is evidence of that the synthetic data well resembles the real data. Further such evidence is displayed in Figure 5.1. It shows that the distribution of the customer journey lengths turned out to be similar after the downsampling of excessive synthetic data. However, the interpretation must be made with caution as these are only two among many possible methods to validate the synthetic data with. On a more detailed level, it is, for example, still unclear whether the distribution of channel types within customer journeys is similar or not.

Regarding the hyperparameters in Table 5.1 for the synthetic data generator, the momentary conversion factor was set to $f_b = 9$ after fitting the data generator. This seemingly high value indicates that users are rather spontaneous in their purchasing decisions, which is confirmed in Figure 5.2, where the importance of touchpoints is shown to increase the closer to conversion the click is.

6.1.2 Real Scenario

After training and testing on real balanced data with 5-fold cross-validation, all models showed roughly the same predictive performance on all metrics. It is noticeable that all models reached the maximum theoretical accuracy within a margin of 0.27 percentage points (p.p.) on the test set. That is to say that the LSTM was of no benefit when only inspecting the mean predictive performance. As argued for in the parallel degree project, the remarkable performance of LTA may be a consequence of the short sequence data (see Figure 5.1) in addition to the relatively high importance of the last click in Figure 5.2 [10].

When studying the mean attributions in Figure 5.3 and the corresponding mean standard deviation in Table 5.5, it is clear that all models yield similar mean attributions for most channels, while LTA, SP and LSTM with attention all have less than a third of the standard deviation of the other models'. At first, it seems counter-intuitive that the standard deviation of the attributions from LSTM with attention is the lowest among models, while LSTM Fractional is expectedly the highest. However, a closer look at Equation 4.4 reveals that the attention attribution is equivalent to the SP attribution weighted by attention and with LSTM predicted instead of true labels.

6.1.3 Hypothetical Scenario

By inspection of Table 5.6, it is apparent that LSTM and LR perform significantly better than LTA and SP in terms of accuracy in the hypothetical scenario. SP performs better only on F_1 and recall, at the cost of a precision close to 50%. This relative improvement of LSTM and LR, in comparison to on real data, may be explained by three new key properties of this synthetic data. Firstly, LSTM and LR have more parameters to learn than the baselines and should benefit from the more than three times bigger dataset. Secondly, the longer customer journeys, illustrated in Figure 5.4, should also benefit the LSTM in accordance with its ability to learn long-term dependencies across sequences as described in Section 2.1.1. Lastly, decreasing f_b from 9 to 0.5 meant that the last touchpoints lost importance, which clearly disadvantaged LTA more. However, it should be pointed out that all models achieve more than 5 p.p. lower accuracy than the theoretical maximum, which is significantly lower than the corresponding value 0.27 p.p. on real data. This shows that also LSTM and LR struggle more in this scenario, yet less than LTA and SP.

6.1.4 Validation of Attributions

The comparison between estimated and true attributions on synthetic data indicates that different models are preferable in different situations.

In the hypothetical scenario with longer sequences and more data, LR, LSTM SHAP and LSTM Fractional seemed to better reflect the true attributions, as shown in Figure 5.5. On the contrary, SP, LTA and LSTM attention all assign positive attributions to LinkedIn, TikTok and Snapchat when their influence on conversion in fact is 0. An explanation for this fault is that those channels still appear in positive customer journeys and these three models will always assign positive attributions to such channels by the nature of their attribution calculation formulas. Hence, the other models are superior when it comes to capturing the causality in this type of data. The higher attribution quality of the more complex models becomes even more evident when inspecting the MAE in Table 5.7. LR, LSTM SHAP and LSTM Fractional achieved significantly lower MAE, although LSTM SHAP and Fractional have the highest standard deviation. Specifically, LR is the best performing model in terms of MAE, while LSTM outperforms all models when it comes to predictive accuracy in the hypothetical scenario. In practice, a high variance could be countered by using the mean attributions after cross-validation.

As opposed to in the hypothetical scenario, SP, LTA and LSTM attention achieved the lowest values on MAE in the real scenario with synthetic data replication. Similar to the discussion in Section. 6.1.3, SP and LTA benefited from short sequences and high importance of the last touchpoint, while not suffering as much from smaller amounts of data.

6.2 Validity of the Results

The validity of the results has been pointed out as an important aspect to consider when conducting experimental studies. The validity may preferably be categorised into four types, i.e., conclusion, internal, construct and external validity [33]. Below follows a discussion about threats touching on these four types of validity.

Threats to conclusion validity include low statistical power or violated assumptions of such tests [33]. Some experiments in this project, such as the TSTR computations, were conducted without cross-validation, which poses a threat to conclusion validity.

The internal validity is threatened whenever there is a risk of influence

on the relationship between observed cause and effect, by an uncontrolled variable [33]. One of the advantages of using a data generator is that the parameters affecting the output can be easily controlled. However, the real data, to which the synthetic data are fitted, have some acknowledged limitations, stated in Section 1.4 and discussed in the parallel project report [10]. Additionally, significant influence from the choice of model hyperparameters can, for instance, not be disregarded since a systematic sensitivity analysis was not done. Weight initialisation of the LSTM is another potential threat, although it was minimised by running multiple trials for most experiments, i.e., using cross-validation.

Construct validity regards whether the results can be generalised to the theory behind the experiment or not [33]. One of the main objectives guiding this study was to measure the quality of attributions. However, as no established theory about measuring attribution quality of LSTMs was found, the chosen method of using a data generator could potentially form a threat to construct validity.

The external validity concerns our ability to generalise the results outside the study [33]. Although the data generator is validated against real data, it can not be concluded whether the hypothetical scenario is realistic or not, nor if the real dataset is representative for this type of setting. Both these factors impact the external validity and must be considered when interpreting the results.

6.3 Conclusions

The results in this project show that the quality of attributions, as measured by MAE, is not unambiguously reflected by the predictive performance of LSTMs. In general, it is not possible to assume a high attribution quality solely based on high predictive performance. This suggests that the ongoing quest for improved accuracy using LSTMs may be questioned. For instance, LSTM achieved the highest accuracy of 64.48% in the hypothetical scenario while also yielding the second highest quality of attributions (0.0161 MAE) using SHAP, only defeated by LR (0.0080 MAE). However, LSTM showed similar predictive performance to the baselines in the real scenario ($\sim 82\%$ accuracy), but LSTM Fractional and SHAP scored the lowest among all models, in terms of attribution quality. Specifically, LSTM Fractional achieved an MAE of 0.0566 and SHAP 0.0311, while LTA reached the best MAE of 0.0058. The attribution quality further varies significantly depending on which attribution calculation method is used for LSTM.

In other words, deep learning models like LSTM with an appropriate

attribution calculation method may be beneficial, depending on the data, when the objective is to accurately model attributions. They demonstrate improved attribution quality in situations with more data and longer customer journeys, where the last touchpoint is not as important. In the opposite situation where the dataset is smaller and contains mainly short sequences, simpler models like LTA and SP seem to work perfectly fine for attribution modelling. The real data used in this degree project is an example of one such situation.

6.4 Future Work

This project has laid the foundations for a change in focus from predictive performance to attribution quality in attribution modelling, using synthetic data. A natural progression is to further develop this data generator along with additional metrics to validate it with. Furthermore, since the LSTM results were sensitive to the choice of attribution calculation method, future research could also focus on constructing new such methods for an LSTM or experiment with new deep learning architectures altogether. Lastly, it would be interesting to find a real dataset, possessing similar characteristics to the synthetic data in the hypothetical scenario, on which to conduct a similar study.

References

[1] Google. Overview of Attribution modeling in MCF. Accessed: 2021-05-30. [Online]. Available: https://support.google.com/analytics/answer/1662518?hl=en

[2] T. Moffett, "The Forrester Wave: Cross-Channel Attribution Providers, Q4 2014," Nov 2014. [Online]. Available: https://services.google.com/fh/files/misc/forrester_cca_wave_q42014.pdf

[3] Google. Conversion. Accessed: 2021-05-30. [Online]. Available: https://support.google.com/analytics/answer/6086209?hl=en

[4] K. Ren, Y. Fang, W. Zhang, S. Liu, J. Li, Y. Zhang, Y. Yu, and J. Wang, "Learning Multi-Touch Conversion Attribution with Dual-Attention Mechanisms for Online Advertising," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, ser. CIKM '18. New York, New York, USA: Association for Computing Machinery, Oct. 2018. doi: 10.1145/3269206.3271677. ISBN 978-1-4503-6014-2 p. 1433–1442.

[5] L. Kakalejčík, J. Bucko, P. A. Resende, and M. Ferencova, "Multichannel Marketing Attribution Using Markov Chains," *Journal of Applied Management and Investments*, vol. 7, no. 1, pp. 49–60, Feb. 2018.

[6] X. Shao and L. Li, "Data-Driven Multi-Touch Attribution Models," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '11. New York, New York, USA: Association for Computing Machinery, Aug. 2011. doi: 10.1145/2020408.2020453. ISBN 978-1-4503-0813-7 p. 258–264.

[7] D. Yang, K. Dyer, and S. Wang, "Interpretable Deep Learning Model for Online Multi-touch Attribution," Mar 2020. [Online]. Available: https://arxiv.org/abs/2004.00384

[8] N. Li, S. K. Arava, C. Dong, Z. Yan, and A. Pani, "Deep Neural Net with Attention for Multi-channel Multi-touch Attribution," Sep 2018. [Online]. Available: https://arxiv.org/abs/1809.02230

[9] B. Dalessandro, C. Perlich, O. Stitelman, and F. Provost, "Causally motivated attribution for online advertising," in *Proceedings of the Sixth International Workshop on Data Mining for Online Advertising and Internet Economy - ADKDD '12*, ser. ADKDD '12. New York, New York, USA: Association for Computing Machinery, Aug 2012. doi: 10.1145/2351356.2351363. ISBN 978-1-4503-1545-6

[10] H. Kindbom and V. Reineck, "Insights on Creating a Growth Machine Using Attribution Modelling," Master's thesis, KTH Royal Institute of Technology, Stockholm, Sweden, Jun 2021.

[11] Publications Office of the European Union. (2016, Apr.) Regulation (EU) 2016/679 of the European Parliament and of the Council. Accessed: 2021-05-04. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[13] S. Squartini, A. Hussain, and F. Piazza, "Preprocessing based solution for the vanishing gradient problem in recurrent neural networks," in *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS '03.*, vol. 5. Piscataway, New Jersey, USA: IEEE, May 2003. doi: 10.1109/ISCAS.2003.1206412. ISBN 0-7803-7761-3 pp. V–V.

[14] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, p. 1735–1780, Nov. 1997. doi: 10.1162/neco.1997.9.8.1735. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[15] C. Olah. (2015, Aug) Understanding LSTM Networks. Accessed: 2021-04-25. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[16] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *arXiv preprint arXiv:1409.0473*, Sep 2014.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, "Attention is All You Need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, New York, USA: Curran Associates Inc., 2017. ISBN 9781510860964 p. 6000–6010.

[18] T. Luong, H. Pham, and C. D. Manning, "Effective Approaches to Attention-based Neural Machine Translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015. doi: 10.18653/v1/D15-1166 pp. 1412–1421. [Online]. Available: https://www.aclweb.org/anthology/D15-1166

[19] S. Lipovetsky and M. Conklin, "Analysis of Regression in Game Theory Approach," *Applied Stochastic Models in Business and Industry*, vol. 17, no. 4, pp. 319–330, Oct 2001. doi: 10.1002/asmb.446. [Online]. Available: https://doi.org/10.1002/asmb.446

[20] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Red Hook, New York, USA: Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[21] S. L. Scott and P. Smyth, "The Markov Modulated Poisson Process and Markov Poisson Cascade with Applications to Web Traffic Modeling," in *Proceedings of the Seventh Valencia International Meeting*, J. M. Bernardo, A. P. Dawid, J. O. Berger, M. West, D. Heckerman, M. Bayarri, and A. F. Smith, Eds. Oxford, UK: Oxford University Press, July 2003.

[22] F. S. Hillier and G. J. Lieberman, *Introduction to Operations Research*, 10th ed. New York, New York: McGraw-Hill Education, Mar 2015.

[23] J. van Kesteren, "Multi Touch Attribution: Searching for the Best Attribution Model," Master's thesis, University of Amsterdam, Amsterdam, Netherlands, Dec 2015.

[24] J. Gaur and K. Bharti, "Attribution Modelling in Marketing: Literature Review and Research Agenda," *Academy of Marketing Studies Journal*, vol. 24, no. 4, 2020.

[25] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic Attribution for Deep Networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, Aug 2017, pp. 3319–3328. [Online]. Available: http://proceedings.mlr.press/v70/sundararajan17a.html

[26] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "Smoothgrad: removing noise by adding noise," Jun 2017. [Online]. Available: https://arxiv.org/abs/1706.03825

[27] K.-K. Tse, "Some Applications of the Poisson Process," *Applied Mathematics*, vol. 05, no. 19, pp. 3011–3017, 2014. doi: 10.4236/am.2014.519288. [Online]. Available: https://doi.org/10.4236/am.2014.519288

[28] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (Medical) Time Series Generation with Recurrent Conditional GANs," Dec 2017. [Online]. Available: https://arxiv.org/abs/1706.02633

[29] J. Jordon, J. Yoon, and M. van der Schaar, "Measuring the quality of Synthetic data for use in competitions," in *Proceedings of KDD Workshop on Machine Learning for Medicine and Healthcare*, ser. KDD '18, Aug 2018.

[30] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *International Conference on Learning Representations*. New York, New York, USA: Ithaca, May 2015.

[31] T. Saito and M. Rehmsmeier, "The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets," *PLOS ONE*, vol. 10, no. 3, p. e0118432, Mar. 2015. doi: 10.1371/journal.pone.0118432. [Online]. Available: https://doi.org/10.1371/journal.pone.0118432

[32] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. 3rd ed. draft*, 3rd ed.

Stanford University and University of Colorado at Boulder, 2020. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/ed3book_dec302020.pdf

[33] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in Software Engineering*, 1st ed. Springer-Verlag Berlin Heidelberg, 2012. [Online]. Available: https://doi.org/10.1007/978-3-642-29044-2