TU WIEN Informatics

Evaluierung der Benutzerinteraktion in der Telekommunikationsindustrie

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieurin

im Rahmen des Studiums

Business Informatics

eingereicht von

Mubashara Akhtar, B.Sc.
Matrikelnummer 1126395

an der Fakultät für Informatik
der Technischen Universität Wien

Betreuung
Betreuer: Univ.Prof. Dipl.-Ing. Dr.techn. Hannes Werthner
Mitwirkung: Mag.rer.nat. Dr.techn. Julia Neidhardt

Wien, 01.11.2019

Mubashara Akhtar   Hannes Werthner

Evaluation of User Interaction in the Telecommunication Industry

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieurin

in

Business Informatics

by

Mubashara Akhtar, B.Sc.
Registration Number 1126395

to the Faculty of Informatics
at the TU Wien

Advisor: Univ.Prof. Dipl.-Ing. Dr.techn. Hannes Werthner
Assistance: Mag.rer.nat. Dr.techn. Julia Neidhardt

Vienna, 1st November, 2019

Mubashara Akhtar | Hannes Werthner

Erklärung zur Verfassung der Arbeit

Mubashara Akhtar, B.Sc.
Laaer-Bergstraße 34-38/4/6, 1100 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 1. November 2019

Mubashara Akhtar

Acknowledgements

First of all, I would like to thank my supervisor Prof. Dr. Hannes Werthner for sharing his expertise and knowledge and for giving me the opportunity to research on this interesting topic. Further, I want to share my sincere gratitude with my co-supervisor Dr. Julia Neidhardt for her support. I would also like to thank both of them for giving me the chance to attend a scientific conference on the topic of recommender systems.

Further, I would like to thank T-Mobile Austria for their collaboration and for sharing their expertise with us.

I would like to express a special thanks to my parents and sisters for their great support and encouragement.

Finally, I would like to thank my beloved husband for always motivating and supporting me throughout this work.

Kurzfassung

Die Telekommunikationsindustrie zeichnet sich durch eine Vielzahl von Produkten und Dienstleistungen aus und Telekommunikationsanbieter haben Zugang zu umfangreichen Benutzerdaten. Das Hauptziel dieser Arbeit ist es, die Benutzer der Telekommunikationsdomäne mit ihren Informationsbedürfnissen und Präferenzen zu verstehen. Basierend auf diese Erkenntnisse, können personalisierte Angebote von Dienstleistungen und Produkten erstellt werden. Recommender Systeme verfolgen das Ziel, ihre Benutzer bei deren Entscheidungsprozessen zu unterstützen indem sie Produkte oder Dienstleistungen vorschlagen, die den Bedürfnissen der Nutzer entsprechen.

Um dieses Ziel zu erreichen und die Benutzer anhand der verfügbaren Daten ganzheitlich zu verstehen, schlagen wir vier verschiedene, sich gegenseitig ergänzende Ansätze vor: Erstens, die Interessengebiete der Nutzer zu extrahieren indem Text Mining auf Chat-Unterhaltungen der Benutzer angewandt wird, welche diese mit dem Chatbot eines österreichischen Telekommunikationsunternehmen durchführten. Zweitens, die Zufriedenheit der Nutzer nach dem Erhalten der Chatbot Antworten zu bestimmen durch Analyse des Benutzerfeedbacks. Drittens, die Stimmigkeit der Chatbot Antworten zu bewerten. Aus den Chat Unterhaltungen werden dafür Sequenzen bestehend aus Fragen der Benutzer und Chatbot Antworten extrahiert, welche im Anschluss analysiert werden. Zum Schluss wird ein Markov Chain Modell mit dem bisherigen Klickverhalten der Nutzer auf der Webseite trainiert. Anhand dieses Modelles wird das Klickverhalten der Benutzer untersucht und ihr zukünftiges Verhalten vorhergesagt. Unsere Ansätze werden mit Daten eines österreichischen Telekommunikationsunternehmens getestet. Das Unternehmen verfolgt das Ziel, die gesammelten Benutzerdaten und die Methoden der Künstlichen Intelligenz zur Entwicklung von innovativen Kunden-Selfcare-Lösungen zu nützen. Ein Chatbot zur Beantwortung von Benutzerfragen auf der Homepage des Unternehmens, wurde bereits entwickelt und ist in Anwendung.

Die Resultate dieser Arbeit können folgendermaßen zusammengefasst werden: Es gelingt uns, Informationsbedürfnisse, Präferenzen und Zufriedenheit der Nutzer anhand der verfügbaren Daten zu analysieren und zu bestimmen. Des Weiteren können wertvolle Informationen zu Telekommunikationsnutzern extrahiert werden durch die Analyse ihre Interaktionen mit der Webseite des Telekommunikationsanbieters. Anhand des expliziten Feedbacks der Benutzer, wird ihre Zufriedenheit mit den Antworten des

Chatbots festgestellt. Anreize für mehr Benutzerfeedback müssen jedoch geschaffen werden.

Schließlich ist es möglich, die zukünftigen Aktivitäten der Benutzer basierend auf ihrem vergangenen Klickverhalten vorherzusagen. Diese Erkenntnisse können zur Erhöhung der Nutzerzufriedenheit genützt werden. Die Struktur der Webseite des Unternehmens muss angepasst werden, sodass die Suchzeit der Nutzer reduziert wird. Es konnte ein Zusammenhang zwischen der Zufriedenheit der Benutzer und einer kurzen Suchzeit auf der Internetseite festgestellt werden. Führt die Suche der User nicht schnell zu einem Resultat, sind die Benutzer frustriert und unterbrechen die Chat Unterhalten bzw. die Suche auf der Webseite.

Im Allgemeinen können die angeführten Methoden und Ansätze für jede Domäne, jeden Chatbot und jeden Clickstream Datensatz angewandt werden.

Abstract

The telecommunication domain is marked by a vast number of products and services on the one side and on the other side, telecommunication providers have access to substantial data. The main goal of this thesis is to address the challenge of understanding users of the telecommunication domain with their information needs and preferences. Based on this knowledge, personalized offers of services and products can be given to users. Recommender systems aim to support their users during their decision making processes by suggesting products or services, which match the users' needs.

To achieve this goal, we propose four different approaches. First, text mining is applied on chat conversations between users and a telecommunication chatbot to determine users' topics of interests. Secondly, further text mining techniques are applied on users' feedback scores and comments to decide if users' are satisfied with the received answers or not. Thirdly, event sequence analysis is applied on sequences of events, which are extracted out of chat conversations to analyze if the predefined chat conversation designs match users' behavior. Finally a prediction model is generated and trained based on users' clicks on the homepage of a telecommunication company. Our approaches are tested using data of an Austrian telecommunication company.

Summarizing the results of this work, it is possible to address the challenge of understanding users' information needs, preferences and satisfaction. Analyzing the chatbot data, invoice/billing, homenet and simcards are determined as users' most popular topics of interests in chat conversations. Moreover, feedback scores given to chatbot's answers and feedback comments in textual form are analyzed. The majority of feedback is negative and considerable differences in feedback for different topics is observed. Finally, the chatbot's answers are evaluated by analyzing the course of conversations after chatbot's answers. Based on the clickstream data, a clickstream model is generated to analyze users' click behavior. Using the test dataset as input to the model, approximately half of the clicks are predicted correctly.

In general, the introduced techniques and models can be applied to any domain, chatbot or clickstream dataset.

Contents

CHAPTER 1

Introduction

1.1 Motivation & Problem Statement

Today's telecommunication companies offer a vast number of mobile products and services e.g. mobile devices, prepaid phone cards, service packages for calling and messaging and Internet services. In comparison to other domains, telecommunication products are marked by three distinguishing features. First, telecom products and services are hard to describe due to their rich number of features. A mobile phone, for example, can be described based on its architecture, camera or software components. Moreover, a service package offers a certain amount of minutes free of charge, has a cancellation period, allows usage in a foreign country or not, etc. Secondly, new telecom products and services are launched frequently. Thirdly, there are strong similarities between products and services of the same provider so that distinguishing them is not always obvious [ZLL+13].

Deciding for the right product or service is therefore not always easy for users of the telecommunication domain. The quality of a decision is positively correlated to the amount of information available for the decision maker as more information allows the decision maker to better understand the decision problem. However, only up to a certain point. Receiving more information after this point, decreases the decision quality as the individual is unable to cope with the huge amount of information. This situation is also known as the *information overload problem*. The additional information is not integrated into the decision making process, yet it becomes more difficult for the individual to set priorities and decide what to choose [EM04].

In the past, skilled salespersons were responsible for recommending matching products and services to users of the telecommunication domain. However, since these salespersons are costly, the telecommunication domain is searching for intelligent information technologies to carry out this task [ZLL+13]. Intelligent information technologies which can be used for this task are recommender systems. As the name suggests, "recommender systems are software tools and techniques providing suggestions for items to be of use to a user" [RRS15]. The purpose of recommender systems is to support their users in their decision making processes. In order to make recommendations, it is first necessary that each system knows its users. Knowing the likes and dislikes of users, as well as their preferences, is essential for generating useful recommendations [JZFF10].

Taking this situation into consideration, it is obvious that the development of a comprehensive user model is an important issue of telecommunication research. In past research, users' CDRs (Call Detail Records) i.e. data that contains telephone calls and messages, have been used for this purpose [HFMO10]. However, today more data sources are available which can provide further insights into users' information needs, preferences and behavior in general.

1.2 Aim of the Work

This thesis aims to model and understand telecommunication users with their information needs, preferences and satisfaction based on real world data gained from digital channels. These channels capture users' interactions with the website of a telecommunication company. The knowledge of users' needs, their preferences and what satisfies them can be used to support them in their decision making process by offering telecommunication services or products, which fit them the best.

The methodological approach of this work can be subcategorized into two parts: chatbot analysis and clickstream analysis. For each of them a separate data source is available. The first dataset captures the interactions between users and the chatbot of a telecommunication business. The chatbot is a conversational agent and communicates with users using a natural language. The chatbot, which is displayed in Figure 3.1, welcomes the users as soon as they visit the website of the company and answers their questions. The aim of the chatbot is to assist the users during their search process on the company's website. The second data source consists of clickstream data. Clickstreams are sequences of HTTP requests, which are generated by users during browsing sessions and traced by the system. These traces allow monitoring users' behavior on a website: which pages the users visit, how long they stay on each page, determining patterns of behavior among user groups, etc.

Chatbot Analysis

Within the scope of this thesis, first the interactions of users with the chatbot of a telecommunication company are analyzed. Based on chat logs, valuable insights on users are gained. The aim is to identify, how chat data can be used to understand users' intentions, problems, preferences and satisfaction. Moreover, we will analyze in more detail which attributes of the chat logs are relevant to attain this aim.

Clickstream Analysis

Secondly, a comprehensive prediction model for the telecommunication area based on clickstream data will be developed. Using the prediction model, users' future actions can be predicted and websites can be adjusted to reduce search processes and increase user satisfaction. The question is, however, whether it is possible to predict users' future decisions (e.g. which product to buy or which service to use) and actions using the proposed model. This also constitutes one of the two main research questions, which should be addressed and solved in scope of this thesis. A data-driven approach will be applied to answer the following research questions:

RQ1 "How can (semi)structured, textual chat data of users be used to model and understand users' information needs, preferences and satisfaction in the telecommunication area?"

RQ2 "How can structured, non-textual website clickstreams be used to predict users' future actions?"

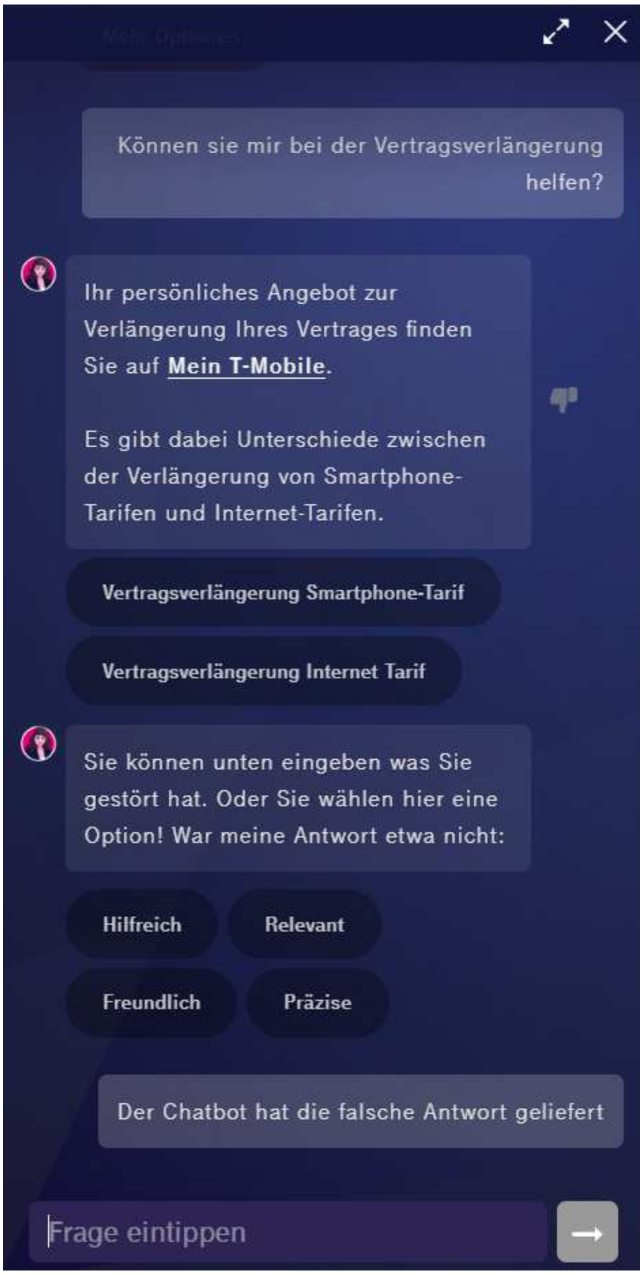

Figure 1.1: An exemplary conversation between the chatbot and an user

1.3 Methodological Approach

The methodology of this work is based on the Cross Industry Standard Process for Data Mining, also known as the CRISP-DM reference model. The process models the entire life cycle of data mining projects and consists of six different phases. The phases are not executed sequentially but allow moving back and forth between them [PCW00].

Next, we cite the specification of the phases as they are described by Chapman et. al [PCW00] and describe each phase in context of this work.

1. **Business understanding**
   a) The first and initial step of the CRISP-DM reference model has the aim to understand the project from a business perspective. Next, the knowledge, which is gained from the business analysis, is transformed into a data mining problem and a roadmap is designed to achieve the objectives [PCW00].
   b) A meeting with the cooperating telecommunicaton company was held in order to understand the background, their objectives and requirements. This meeting was followed by a literature review as an essential step for understanding the domain, as well as current projects and research in the area. Moving back to the phase of business understanding was necessary a few times during the entire process for removing ambiguity or discussing questions, which emerged.
2. **Data understanding**
   a) In the "data understanding phase" first the data is collected and different steps are realized to become familiar with the data. Quality problems are highlighted, as well as first insights into the data are gained [PCW00].
   b) The data understanding process started with receiving the data from the cooperating partner. Two datasets were provided: the first one capturing click-stream data and the second dataset consisting of chat data. First, explorative analyses were conducted to get insights into the datasets. Feedback on the quality of the data was returned to the partner.
3. **Data preparation**
   a) The next phase is the "data preparation phase", which includes tasks necessary for constructing the final dataset. During this phase the data is tabled, recorded, transformed, cleaned and relevant attributes are selected. The order and selection of these tasks depend on the dataset [PCW00].
   b) The two datasets were reviewed, formatted and relevant attributes for the analyses were selected. At the end of this phase we had test and training datasets for the subsequent phases. The intermediate results of explorative analysis and data preparation were presented and discussed.
4. **Modelling**
   a) In the "modelling" stage, techniques and methods for modelling are selected and the models' parameters are calibrated. It is often possible to use different methods for solving the same problem. Shifting back to the data preparation

phase is often necessary and depends on the selected techniques and its requirements [PCW00].

b) The phase of modelling was accompanied by several discussions on possible models and their development. Analysis and modelling was done using the programming language R. For the purpose of visualization other tools e.g. Tableau, Gephi and Dataiku were used.

5. **Evaluation**

   a) In this stage the model is built and evaluated. Before deploying the model, it is essential to adjust the model based on the evaluation. It is reviewed whether the business objectives are achieved or not. This phase is completed with a decision on the use of the data mining results [PCW00].

   b) The clickstream prediction model was evaluated on three different levels: First, the model was evaluated statistically. Secondly, the model was evaluated with the aid of domain experts. As a characteristic of success we defined the applicability of the model in order to predict users' decisions based on their past behavior. Finally, the predicted user behavior was compared to real world data capturing the user's actual behavior. The insights gained out of the chat data analysis were evaluated and discussed with the support of the data provider, who is an expert of the telecommunication domain.

6. **Deployment**

   a) The model creation is not the final step of the process as the results have to be presented in a way to the user that they are usable and applicable for the user. The "deployment" step often involves an application of the model for decision making processes. The extend of the deployment stage depends on the business and its requirements. Depending on the project, the deployment phase can be e.g. the generation of a report or the implementation of certain processes and actions within the organization. Otherwise the gained knowledge is not applied [PCW00].

   b) During the final phase, the developed models and gained insights were presented to the telecom partner, suggestions for acting upon the results were provided and a project report was prepared.

1.4 Outline of the Work

The structure of this master thesis is organized as follows: Chapter 2 gives an overview over the state of the art in recommender systems, user modelling, dialog systems and clickstream analysis. Chapter 3 deals with chatbot analysis and evaluation. The dataset is described as well as the conducted preprocessing steps. The chapter also describes text mining and event sequence analysis - how it is conducted using the data and an evaluation of the results. Chapter 4 presents the clickstream dataset, the preprocessing steps conducted, the modelling of Markov chains and predicting based on the Markov chain models and the testset. In the fifth chapter we discuss the insights from the preceding chapters. Furthermore, the results are reviewed with respect to their general applicability. The sixth and final chapter concludes the thesis with a short summary, outlining its limitations and highlighting future work.

CHAPTER 2

State of the Art

This chapter gives an introduction to the state of the art on two main topics of this thesis: dialog systems and clickstream analysis. First, an overview over the field of recommender systems is provided and the purpose of these systems is pointed out. Next, the importance of understanding users, as well as their behavior, needs and preferences for delivering appropriate recommendations is discussed. As it is stated in the introduction of this thesis, understanding users with their information needs, preferences and satisfaction, is one of the research questions of this work.

Within the scope of this thesis, the area of chatbots is analyzed based on the digital agent of an Austrian telecommunication company. The agent aims to suggest the best fitting channel (FAQs, live chat, forum) to users for finding answers to their questions and topics of interests. Before analyzing the chatbot's data in more detail in Chapter 3, section 'Dialog Systems and Chatbots' gives an introduction to dialog systems and chatbots, industrial chatbots and the applicability of chatbots as recommender systems.

Finally, in section 'Clickstream Analysis', clickstream analysis is introduced and different models for analyzing users' click behavior are presented. Additionally, an overview of state of the art research in the area of clickstream analysis is presented.

2.1 Recommender Systems

Burke [Bur07] defines recommender systems as "personalized information agents that provide recommendations". These suggestions have to be useful for the users, for whom the recommendations are given. The result of a recommender system is an option, which is personalized for a certain user or user group. Objects, which are recommended to the user e.g. a book or a restaurant, are referred to as "items" [RRS15].

Due to the abundance of information, products and services on the web, users find it more difficult to make choices. Schwartz [Sch04] describes this as a "paradox of choice" since more highly relevant options lead to poorer choices and satisfaction. The possibility of choice is mostly beneficial but too much choice causes the user to become overwhelmed and potentially decreases well-being.

Recommender systems have proven to be useful for handling the information overload problem [RRS15]. Well-known companies such as Netflix, Amazon, LinkedIn and TripAdvisor have integrated recommender systems and focus on providing personalized products and services to their users. Moreover, these companies work on developing new methods to further develop and improve of these systems.

Recommender systems have three main components: background data, input data and algorithms. Background data is information the system already possesses before the recommendation process starts. This is often knowledge about the field of recommendation e.g. genres of books or items in an online shop. Input data is data about the specific user who interacts with the system and the algorithms use the background data and input data available for generating recommendations [Bur02].

Understanding and Modelling Users

As it is mentioned in the previous section, user data is a key input for recommender systems. In order to address users' needs, it is first of all necessary that each system knows its users. Understanding the users, their needs, behavior and preferences helps businesses to tailor their products, services and strategies according to their customers. Modelling, profiling, analyzing and understanding users becomes increasingly important in many different industries and a key to success in today's data driven world [HBZG14]. The aim of user modelling is to capture the users by encoding their preferences and needs [RRS15]. Past research has found significant correlations between the personality of a user and the user's preferences in a number of domains including music, travelling and language [RC97, TC15, NSSW14]. These correlations can be successfully exploited as long as the system has sufficient information about its users [TC15].

User information can be gathered explicitly by conducting questionnaires or implicitly using machine learning techniques. In the past, it was only possible to capture information about the personality of a person after completing extensive questionnaires. This expensive process of data collection was an obstacle for the development of recommender systems. Though, in recent years valuable sources of personality assessment have been found e.g. social media streams such as Facebook and Twitter [TC15]. The picture-based approach for eliciting users' preferences, which is introduced by Neidhardt et. al [NSSW14], acquires users' travel profiles based on picture selection. Users select pictures, which are then mapped onto predefined seven factors of touristic behavioral patterns. For each user an individual profile is generated out of these seven factors [NSSW14].

User Interaction in the Telecommunication Domain

A key characteristic of the telecommunication domain is the large amount of customer data available. Telecommunication providers know how long their customers' phone calls take, to whom they talk to, which websites they browse, which mobile phones they own, etc. Based on this data further information about the user can be extracted. Moreover, the data which is available to telecommunication providers is very divers in its structure as they have access to call detail records, demographic data, webtracking data and smartphone data.

The user data used in scope of this work is captured implicitly. First, the chat histories of the chatbot's conversations with users are recorded. Secondly, users' click behavior is tracked during their activities on the telecommunication company's website using a web cookie. Using this data, users are modelled based on their preferences and interests on the one side, and users' future steps are predicted on the other side. It is worth mentioning that the extraction of valuable information out of the available data sources is still in an early stage in the telecommunication area compared to other, well-established fields such as healthcare or travel.

State of the art research which uses telecommunication data to gain insights into users' characteristics and behavior are presented next. In scope of their study, Haddad et. al [HBZG14] have utilized mobile data to predict the behavioral patterns of individuals. The study proves that it is possible to detect short term daily patterns as well as long term e.g. monthly or yearly routines of users based on their data. Another study based on telecommunication data was conducted by Mit et al. [MWHZ13] where they classify users' "contacts according to life facet (family, work, and social)". The classification was done used communication logs and phone logs between the users and their telephone contacts, as well as sensor data gathered from smartphones and social media. Kong et. al [KZM16] use call detail records to analyze users' mobile usage patterns and online behavior.

The previously mentioned papers use mobile phone and telecommunication data to extract insights about users' characteristics, behavior and their environment. Though, the development of intelligent tools such as recommender systems based on the gained insights, is still a mainly unexplored area. Pioneering work has been done by Zhang et. al [ZLL+13] as they have developed a personalized recommender system to suggest telecommunication products and services to customers. The authors deal with the difficulty of suggesting products and services, which have complex features on the one side and are updated frequently on the other side.

2.2 Dialog Systems and Chatbots

The use of computers for answering textual questions goes back to the early 1960s, when systems implementing question answering algorithms were first built [MJ09]. In this section, first, the role of chatbots as dialog systems for interacting with users is discussed. Next, two common paradigms used by computational systems for question answering are presented, as well as different types of dialog systems. Finally, an introduction to industrial chatbots is given and the telecommunication chatbot, which is analyzed in this work, is introduced.

Chatbots - an Interface for Communicating with Users

Chatbots which are used for interacting and communicating with users can be characterized by a number of attributes. In the following, these characteristics are listed and described.

1. Chatbots as conversational systems are critique-based systems. The chatbot has the possibility to ask the user if the recommendation was useful or not. Based on users' feedback, the user model can be adapted and in future more fitting recommendations can be given [IAKH18].

2. Chatbots can cope with two issues, which might appear while retrieving information for users' requests by asking the user further questions. First, in case that no matching recommendation can be found due to lack of information about the user and his preferences. Secondly, if the number of retrieved recommendations is too high to present it to the user. In this case the system can ask the user further questions to limit the number of generated answers [NTW18].

3. Similar to recommender systems, chatbots have to deal with the cold start problem. Based on the proposed questions, the system tries to learn as much about the users as possible without having any information about users' background. With every input from users' side i.e. questions or feedback, the user model can be updated and improved.

Underlying Question Answering Paradigms

This section introduces two main paradigms for question answering: information-retrieval-based question answering and knowledge-based question answering. The chatbot which is analyzed in this thesis, has an underlying knowledge-based question answering system.

1. Information-retrieval-based Question Answering

Information-retrieval-based question answering aims at answering user's questions by finding short texts in a collection of documents available to the system e.g. the Web, which contains possible answers to the proposed questions. This paradigm strongly relies on information availability on the Web in form of a vast collection, which can be searched for the answer [MJ09].

The process of information-retrieval-based question answering consists of three phases: First, the processing of the question. Next, the matching passage is retrieved from the information source and ranked. As a final step, the answer is processed [MJ09]. This process is visualized in Figure 2.1.

The question processing step has the goal of extracting information based on the proposed questions. Using this information, the answer type can be determined. The answer type defines, for example, that the user is looking for an answer of type person or location. Moreover, the query contains a list of keywords, which are used for finding the right answer. During the passage retrieval step, the generated query is sent to an information-retrieval system, which returns a set of documents. Next, non-relevant passages are filtered out and the remaining passages are ranked. The ranking of the passages is commonly done by supervised learning methods. The final phase of question answering is the answer processing step. Out of the ranked passages the answer is extracted. The extraction can be performed based on a pattern-extraction algorithm, which analyze each sentence of a passage based on a list of features [MJ09].

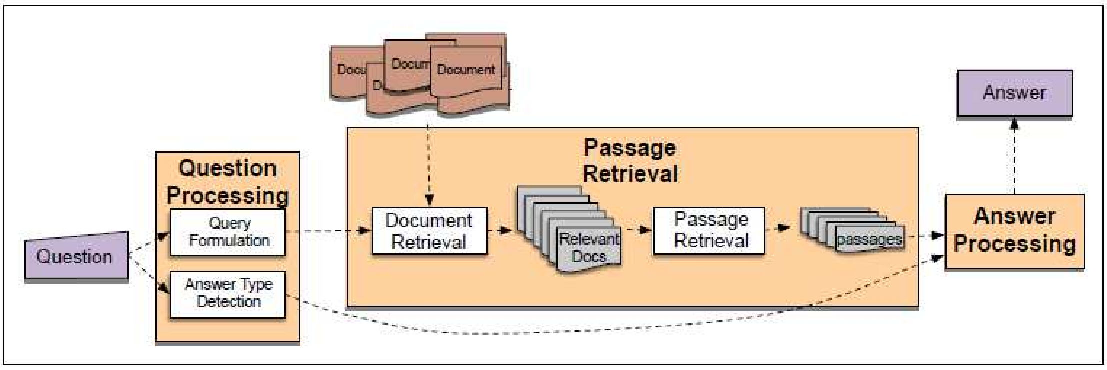

Figure 2.1: Information-retrieval-based question answering process [MJ09]

2. Knowledge-based Question Answering

Knowledge-based question answering systems (KB-QA) answer questions in a natural language using a structured database. The database can be either a full relational database or a more simpler database e.g. a set of RDF triples. Systems which map a text string to a logical form for example a query language are called *semantic parsers* [MJ09]. Table 2.1 displays exemplary logical forms, which are produced by a semantic parser to answer questions.

Question	**Logical Form**
"When is the new IPhoneX available for sale?"	sale-start-date (IPhoneX, ?x)
"Which is the cheapest telephone tariff?"	$\lambda x.tariff(x) \wedge cheapest(x)$

Table 2.1: Logical forms by a semantic parser [MJ09]

For frequently asked questions, rule-based methods are convenient as simple rules can be written for often occurring questions. Systems using a rule-based method have a knowledge base consisting of facts and rules. A rule can be interpreted as a condition. It is expected that if a certain condition is fulfilled, the consequence is true as well i.e. if a user asks about the new IPhoneX, display the site "www.t-mobile.at/iPhone_X" to him.

Supervised methods are also used for question answering in KB-systems. The system has supervised data which maps a set of questions to their correct logical form. This data is then used to find the logical form for new questions.

Types of Dialog Systems

Dialog systems, which are also known as conversational agents, are systems designed for communicating with users using a natural language. These systems can be sub-categorized in two sections: task-oriented dialog systems and non-task-oriented dialog systems [MJ09].

1. Task-oriented dialog agents

Task-oriented dialog agents are systems built for a certain purpose and for conducting short conversations. Well-known digital agents e.g. “Siri” and “Alexa” are task-oriented dialog agents, which are designed for simple tasks such as making phone calls, describing routes or finding restaurants. Conversational agents, which are installed on companies' websites to assist customers with their problems and to answer their questions, are also task-oriented dialog agents [MJ09].

These agents are often based on domain ontology, which is a knowledge structure defining frames. Each frame has a set of slots which should be filled with users' inputs. Therefore these systems are also called frame based dialog agents. The slots determine the information the systems needs in order to answer certain questions. For answering questions about telephone tariffs, the following slots can be defined e.g.: costs, commitment period, mobile phones and free minutes [MJ09]. Table 2.2 represents questions the system can ask the user to get information for filling the slots.

Slot	**Question**
Costs	“Which price are you willing to pay for the tariff?”
Commitment Period	“How long should the maximum commitment period be?”
Mobile Phone	“Which mobile phone do you wish?”
Free Minutes	“How many free minutes should your tariff include?”

Table 2.2: Frame-based question answering system

2. Non-task-oriented dialog agents

Non-task-oriented dialog agents are used for longer and more complex conversations with the purpose of imitating conversations between humans. These systems don't focus on a certain task but are meant for entertaining users. ‘XioaIce’ developed by Microsoft Peking is an example for a non-task-oriented dialog agent [Boi18]. The chatbot is designed more like a friend. On the one side she can answer questions e.g. about the weather or news and on the other side she has the ability to react to the emotional states of individuals talking to her. Therefore, many users reported that they did not recognize in the first ten minutes of the conversation that Xioaice was a chatbot [LFB+17].

In research non-task-oriented dialog agents are sometimes also referred to as chatbots. In literature various explanations for the term ‘chatbot’ can be found. Chatbots have been used in various contexts, for task-oriented and non-task-oriented systems [Boi18]. This naming distinguishes itself from the media and industry, where the term chatbot is often used as a synonym for task-oriented conversational agents e.g. the chatbot analyzed in this work.

Industrial Chatbots

Many companies have recognized chatbots as “the next big thing” in terms of customer relationships. In today’s digital age customer relationship is shaped by an empowerment of customers due to increased information availability, digital communication channels and more diverse possibilities for reaching customers [KH18]. Regarding the usefulness of chatbots from customers’ point of view, a chatbot enables 24-hour customer service, personalized interaction and no waiting time. For companies chatbots entail time and cost savings as many processes can be automated and employees can be appointed to more complex tasks [KH18].

Kawohl et. al [KH18] analyzed German stock companies in regard of their chatbots. Out of 80 companies, 12 companies (15%) used a chatbot for customer communication at the time of the study. Four companies belong to the area of mechanical engineering or transport. Furthermore, two financial companies and two commercial companies have also integrated chatbots. The remaining four companies belong to these areas: pharmacy, computer engineering, electric and news service.

One out of the previously mentioned 12 companies is T-Mobile Austria, the telecommunication company providing data for the analysis done in scope of this thesis.

The company has implemented a chatbot named “Tinka” on it’s website, which welcomes users as soon as they visit the site and answers their questions. The term Tinka is a byword for “T-Mobiles interaktive neue Kommunikations-Assistentin”. This chatbot can be classified as a task-oriented dialog agent which aims to assist users during their search process on the company’s homepage. Therefore the chatbot answers simple and common user questions directly on the one hand and directs them to other question answering channels on the other hand. The chatbot is a knowledge-based question answering system using the non-relational HBase database for short-time storage and the Hadoop distributed file system (HDFS) for long-time storage. Answers to users’ questions are retrieved using different query languages to map users’ questions to the matching answers.

2.3 Clickstream Analysis

This section gives an introduction to the area of clickstream analysis and presents three different models for clickstream analysis used in past research: click sequence models, time-based models and hybrid models. Furthermore, Markov chains are presented as a modeling technique for analyzing and predicting clickstreams.

Clickstreams can be defined as “traces of click-through events”, which are generated by users of a website during each browsing session and traced by the system. A clickstream is a sequence of HTTP requests a user sends to a website while interacting with the site [WKW+13].

Being able to model and analyze an user's web browsing behavior can open up many new opportunities for electronic commerce. Analyzing clickstream data facilitates personalized recommendation and advertisement. Clickstream data contains explicit information about the user as well as implicit information. Explicit information is information the user communicates himself with the website e.g. name, age, address and gender. Whereas implicit information e.g. clicking behavior can provide important insights about a person by utilizing data mining techniques e.g. shopping behavior or how much time a user needs for taking a decision [WKW+13].

Clickstream analysis has been used in various field, e.g. for the purpose of sybil detection, analyzing customers' shopping behavior or learning their interests [WKW+13, SKN05, GS00]. In past years, researchers have analyzed the behavior of users within a web page by analyzing the generated clickstream data, e.g. the time spent on each particular page and between websites, the order and amount of sites visited [SKN05].

Modelling Users' Click Behavior

Wang et. al [WKW+13] propose three different models for modelling clickstream data: click sequence models, time-based models and hybrid models. Next, each model will be described briefly and its usage will be demonstrated by an example.

1. Click sequence model: This simple model only considers clicks and interprets each single clickstream as a sequence of clicks the user did on the site. The time spent on each site is ignored thereby and the clicks are ordered according to their order of occurrences. A click sequence model for clickstream analysis is developed by Su et. al [SYLZ00]. The authors present a model, which they name a *n-gram prediction model*. Based on sequences of clicks, each sub-session of length n out of the entire session is a n-gram. These n-grams are scanned once by the algorithm and used afterwards for prediction. The basic functionalities of the model can be demonstrated by the following example:

   Using the sessions $ABCD, ABCF, ABCF, BCDG, BCDG, BCDF$ as inputs for the model and setting $n = 3$, the model calculates the hash table presented in Table 2.3. Receiving a session for prediction, this table is scanned to predict the next step of the session.

N-Gram	Prediction
A,B,C	F
B,C,D	G

   Table 2.3: Example for a n-gram prediction model [SYLZ00]

2. Time-based model: Time-based models capture the time of each event occurrence. In this case a clickstream is represented as a list of arrival times of the events:

$[t_1, t_2, ..., t_n]$. Chatterjee et. al [CHN03] develop a time-based model for analyzing users' responses to websites' ads. Their model captures the time when a user visits the website to determine how much time has passed between the first and the second visit. Depending on the time factor, it is determined how likely users will react to banner ads.

3. Hybrid Model: As the name already indicates, hybrid models consider the click type as well as the arrival time of each click event. For developing their clickstream model Lee et. al [LPSH01] consider the sites a user visits within a session as well as the timestamp of each click event. User events are assigned to one of four general shopping events: product impression, clickthrough, basket placement and purchase. This simplifies the entire shopping process and allows developing a hybrid model, which considers the event types and the time component.

The previously described studies give an insight into the vast possibilities of analyzing clickstream data in order to understand telecommunication users, their behavior, information needs and preferences. One of the presented models, the click sequence model is selected in scope of this thesis for modelling the provided clickstream data. The aim is to focus upon the websites a user visits and the order of visiting them within one clickstream session. The amount of time spent on each single website is ignored thereby.

Modelling Techniques

Modelling our data as a click sequence model, requires as a next step the decision which modelling technique to choose. We decide to use Markov chains for modelling our clickstream data. These models have been well suited for this modelling problem in past research [GÖ03]. Markov chains can be used as models for data, which consists of sequences of states or events with or without timestamps. In case of our clickstream data these states are represented by the clicked URLs. Markov chains are probabilistic model structures, which are used to describe and analyze probabilistic processes [GRM15]. In Chapter 4 the characteristics of Markov chains and how they can be used to model clickstream data are described in more detail.

CHAPTER 3

Chatbot Analysis

The goal of this chapter is to model and understand users' information needs, preferences and satisfaction based on chat data collected by the chatbot of a telecom provider. Using this knowledge, the chatbot can be improved in future in order to align to user behavior.

In this chapter the following research question is discussed:

RQ1 "How can (semi)structured, textual chat data of users be used to model and understand users' information needs, preferences and satisfaction in the telecommunication area?"

To answer this question in greater detail, the following sub-questions are defined and discussed:

RQ1.a "Which user interaction attributes are relevant and determinative for fulfilling this purpose?"

RQ1.b "How can explicit, textual feedback and inexplicit information extracted from the chat data be utilized in order to determine users' satisfaction?"

RQ1.c "How can non-textual information gained from the chat data be used for determining whether users' information needs were satisfied or not?"

As it has been mentioned previously in Chapter 2, in the past the term chatbot has been used for different types of systems in literature and industry. The referenced chatbot of this work is a task-oriented, knowledge-based system, which is built for customer support as it assists telecommunication users in their search processes. Similar chatbots are used for the same purpose by companies like *CNN*, *AirBnB* and *Spotify*.

First, the telecommunication domain and the available datasets are described in section 3.1. Furthermore, the different data preprocessing steps, which are conducted in order to prepare the dataset for further analyses are presented in section 3.2. In order to answer the research questions, text mining is applied on the chat logs in section 3.3.

Users' feedback comments and scores are analyzed for determining users' satisfaction. Finally, event sequence analysis as a method for converting textual chat data into a more structured form is presented in section 3.4.

3.1 Overview & Descriptive Analysis

This thesis addresses the needs, preferences and satisfaction of users in the telecommunication domain. Based on users' information needs, more personalized services and products can be provided to increase satisfaction. The telecommunication domain is marked by offering a huge number of products and services to their customers. From service packages for calling, messaging and Internet surfing to numerous devices e.g. mobile phones and smart watches. In Chapter 1 three characteristics of telecommunication products and services are mentioned:

1. High number of features
2. Regular launches of new products and services
3. Similarity of products and service packages

Taking a closer look at the homepage of our cooperation partner, an Austrian telecommunication company named T-Mobile Austria GmbH, the following services and products can be found:

- Service packages for mobile phones (calling, messaging, Internet)
- Service packages for Internet at home
- Prepaid phone cards
- Mobile phones
- Tablets
- Smart living devices e.g. car connect and smart watch

Additionally, customer support for the listed products and services is necessary and provided. Analyzing the FAQ page, topics such as billing, network availability, contract agreement, roaming and locked sim cards become apparent. The goal is to improve customer satisfaction by providing sufficient self-care solutions e.g. chatbots. The chatbot data used for training algorithms and models in this chapter covers conversations between users and a telecommunicaton chatbot between February and August 2016.

For each single month between February and August 2016, a separate datafile is provided, which captures the interactions of each particular month. Some of the presented analyses and results refer to single months, e.g. when we aim to observe changes across different months and compare them to each other. All other analyses are conducted based on the entire data from February to August 2016, merged into one single dataset. This dataset will be referred to as X_{Tinka}. The datasets for the different months are named according to the months: $X_{February}, X_{March}, ..., X_{August}$.

The interactions between users and the chatbot are described by **32** different attributes. In table 3.1 the ones are listed, which are relevant for our analyses. The attributes are grouped into three categories: basic attributes, attributes used for feedback analysis and attributes used for event sequence analysis.

Category	Attributes
Basic Attributes	*interaction ID, session ID, timestamp, user ID, dialogname (= category the dialog is assigned to)*
Feedback Attributes	*feedback label, feedback score, feedback comment*
Event Sequence Analysis Attributes	*previous page, interaction type, linkclick url*

Table 3.1: Attributes of the chatbot dataset

An Exemplary Chatbot Conversation

The chatbot analyzed in this thesis is named "Tinka". Tinka is a self-care solution, which welcomes users at the homepage of a telecommunication company as soon as the site is visited. The chatbot answers users' questions directly or directs them to other service channels e.g. a customer service employee or the FAQ site where they can find answers to their questions. The aim of the company is to increase user satisfaction and to reduce costs by engaging less employees for customer service. The chatbot extracts the answers to the questions out of the "Tinka brain". The Tinka brain is the main component of the chatbot's architecture where data is collected and stored. Using an API the chatbot can access data which is necessary for answering users' questions.

Before going into more detail by discussing each single attribute of the provided dataset, an exemplary chatbot conversation is presented in Figure 3.1. This conversation starts at the starting page of the company's website *(https://www.t-mobile.at/)*. Analog to all other conversations, this chat has an unique session ID and it consists of five different interactions. First, a question is proposed by the user ("Können Sie mir bei der Verlängerung helfen?" - "can you help me with the extension?"). Next, the chatbot gives an answer and provides a link for further information. The link *https://mein.t-mobile.at/myTNT/start.page* gives the user the possibility to exit the conversation and find the answer on the website of the company. The third interaction is user feedback given to the chatbot. The user has the possibility to rate the answer positively by choosing an upward thumb or negatively with a downward thumb. After choosing the downward thumb as the fourth interaction, the user is asked for a feedback comment at the end of the conversation. The fifth and final interaction is a feedback comment by the user: “Der Chatbot hat die falsche Antwort geliefert” (“The chatbot delivered the wrong answer”).

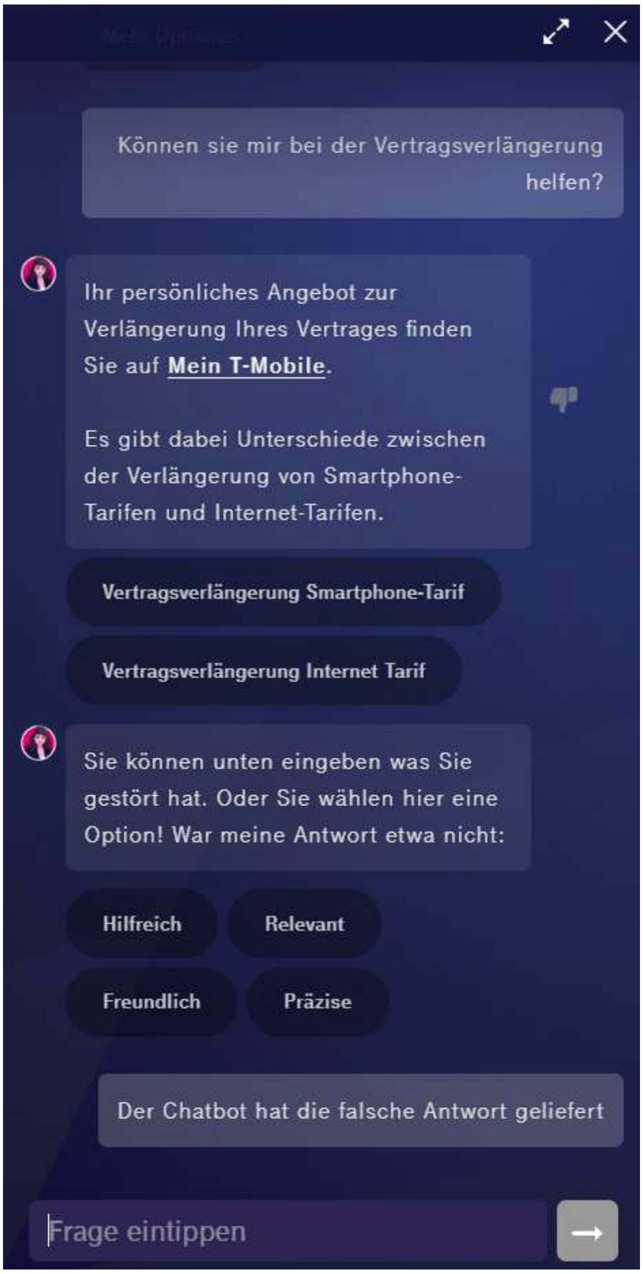

Figure 3.1: An exemplary conversation between the chatbot and an user

3.1.1 Basic Attributes

Starting with the category of basic attributes, these attributes provide an overview and basic understanding of the available data and allow grouping the data according to chat sessions, time, categories or users.

Interaction ID & Session ID

The attributes interaction ID and session ID are essential in respect to grouping the entire data. The dataset X_{Tinka} contains **673,176** entries. Each entry of the dataset represents an interaction i.e. a question asked by the user, an answer given by the chatbot or feedback provided by users. The data can be sub-categorized into sessions and interactions.

The listing below displays the number of interactions per month:

Month	total Number of Interactions
February	91,293
March	103,027
April	80,638
May	86,576
June	89,851
July	113,533
August	108,258
Feb-Aug	**673,176**

Table 3.2: Number of interactions per month

Each interaction is allocated to a session. In sum our dataset X_{Tinka} captures **215,859** different sessions.

User ID

The user ID is a hashed contract ID provided by the Single-Sign-On (SSO) token as soon as users sign in on the website. User identification would allow us to track and analyze user behavior across different sessions. Identifying users also allows to consider background information of them e.g. age and contract details while discussing users' behavior, preferences and needs.

Timestamp

The timestamp attribute captures various aspects of sessions and interactions e.g. the duration of sessions, when the sessions start and end, as well as the distribution of sessions across weekdays. It facilitates analyzing the workload of the chatbot at different times of the day and different days of the week.

The following Figure 3.2 shows the amount of open sessions at a specific time of the day. For this plot the conversation data from February to August is aggregated (X_{Tinka})

according to the starting points of the conversations. A peak of chatbot interactions can be found between 9 am and 10 am.

Figure 3.3 displays the distribution of chat sessions among the different days of the week. The biggest amount of interactions (18.2%) took place at the beginning of the week (Monday and Tuesday). The interactions on weekends are lower: 8.8% on Saturdays and 10.4% on Sundays. A pie chart of the chat sessions according to weekdays can be found in Figure 3.3.

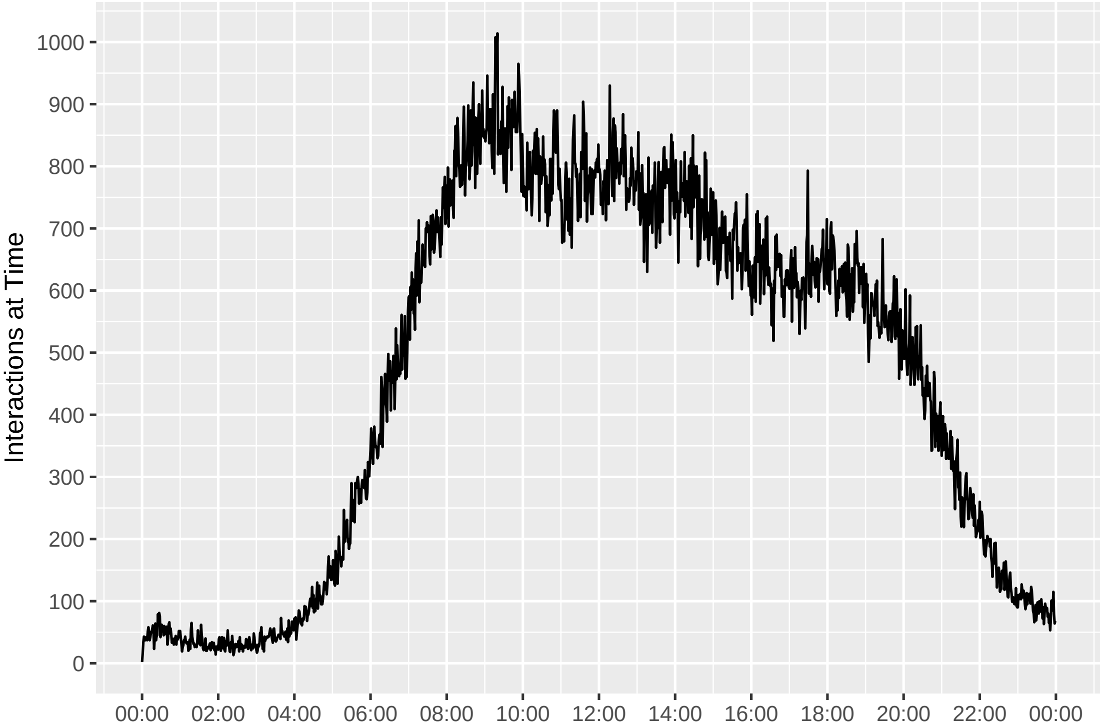

Figure 3.2: Plot of active sessions at a certain point of time (Feb - Aug 2016)

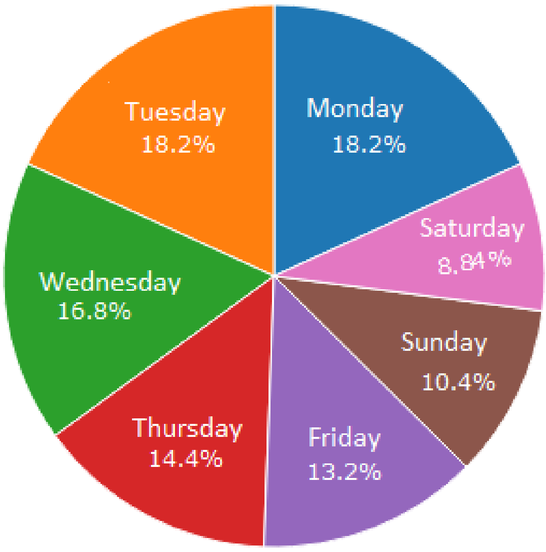

Figure 3.3: Count of interactions according to weekdays (Feb - Aug 2016)

The longest session during the entire timeframe February - August 2016 lasted for 138 minutes. In total 122,540 sessions are shorter than one minute as these conversations end within seconds. As these sessions only last for some seconds, the median value of session duration in minutes is equal to zero for each month. Table 3.3 displays the duration of the longest session for each month, the number of sessions, which have a duration shorter than one minute and the total number of chat sessions per month. The median value for each month is equal to zero as more than half of the sessions of each month are shorter than one minute as it can be seen in column 'Shorter than one min'. A chart of all session durations can be seen in Figure 3.4.

Month	Total Sessions	Longest (in min)	Shorter than one min
February	31,610	81	18,705
March	33,041	138	18,646
April	26,375	52	14,737
May	27,921	68	15,569
June	28,622	126	16,175
July	34,910	86	19,431
August	33,380	60	19,277
Feb-Aug	**215,859**	**138**	**122,540**

Table 3.3: Sessions per month

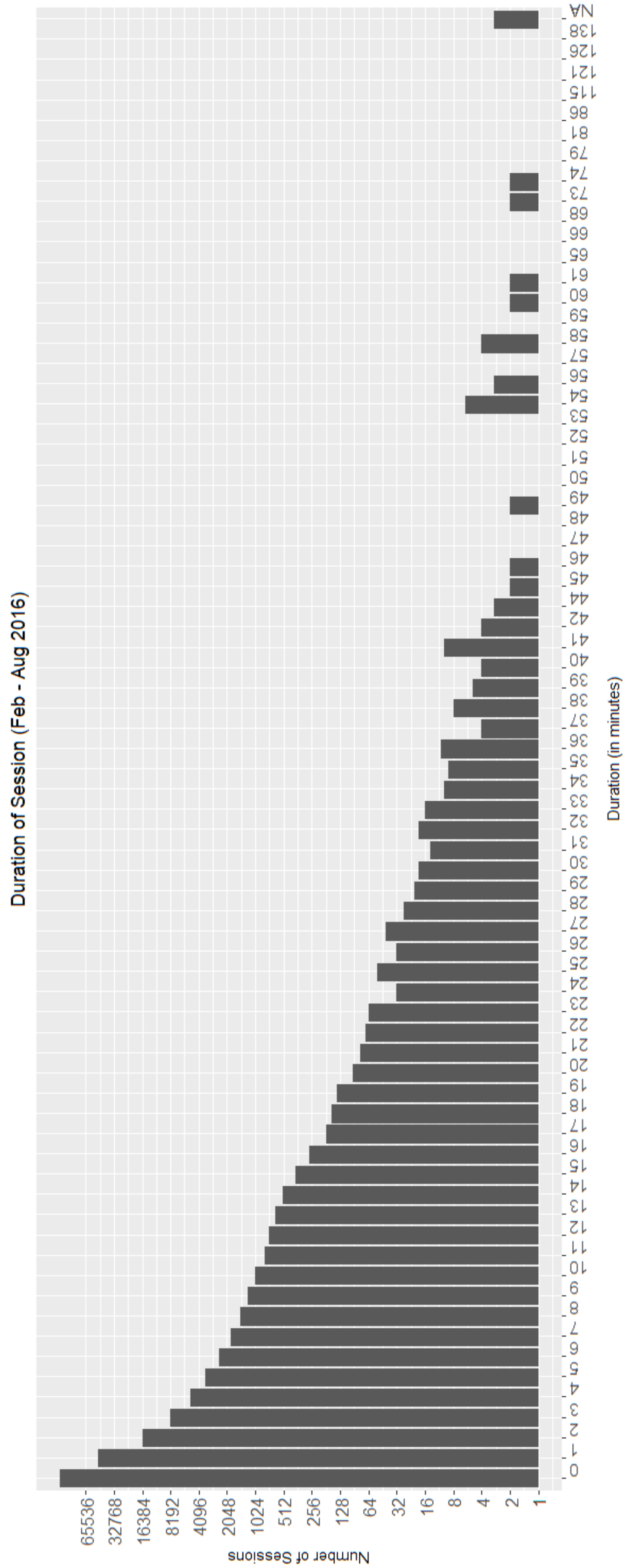

Figure 3.4: Duration of chat sessions (Feb - Aug 2016)

Dialogname (Categories)

Another basic attribute of the dataset is the dialogname, which we will refer to as 'category' in further consequence as this attribute represents a labelling of the chat dialogs to a certain category, e.g. billing, Internet or roaming. The labelling itself is done by the telecommunication company, which provided the dataset. The company labeled a small part of all interactions with a label, concretely 101,115 out of 673,176 interactions are labeled.

As each interaction is part of a chat session, next the categories of interactions are assigned to the sessions they occur within. The majority of sessions has exactly one labeled interaction and it is possible to assign a specific category to these sessions. In total 34,651 chat sessions out of 215,859 can be labeled with a certain category. We utilize this labelling to assign each chat session to a certain category e.g. 'Homenetbox', 'Rechnung' or 'Email'.

In case of 1,779 sessions, different interactions of the same session are labeled with different categories. An exemplary chat session with different categories is displayed in Figure 3.5. The session starts with a question of the category 'Rechnung' and carries on with the topic 'Homenetbox'. In this case it is not possible to assign an exact category to the chat session. These 1,779 sessions are removed during preprocessing as it is not possible to assign them to a particular category.

In total 64 different categories exist to whom sessions can be matched. The top categories are listed in Table 3.4. The knowledge of the popularity of categories can be beneficial for future improvements and development of the chatbot. Table 3.4 also displays the number of sessions out of the dataset X_{Tinka} assigned to a certain category. The low amount of categorized sessions must not be affiliated to a faulty labelling. As it is mentioned before, 34,651 sessions can be labeled exactly and only 1,779 sessions have to be excluded because of ambiguousness. Matching sessions during analysis to a certain category is not possible as only the name of the category is available and no further information, which is necessary to decide if a session is part of a certain category or not. The category with the highest amount of chat sessions is "Email", followed by "Homenetbox" and the category "Rechnung2". The category "Rechnung" is sub-categorized in "Rechnung" and "Rechnung2". In total, these two subcategories ("Rechnung", "Rechnung2") have the biggest share of 6,575 chat sessions.

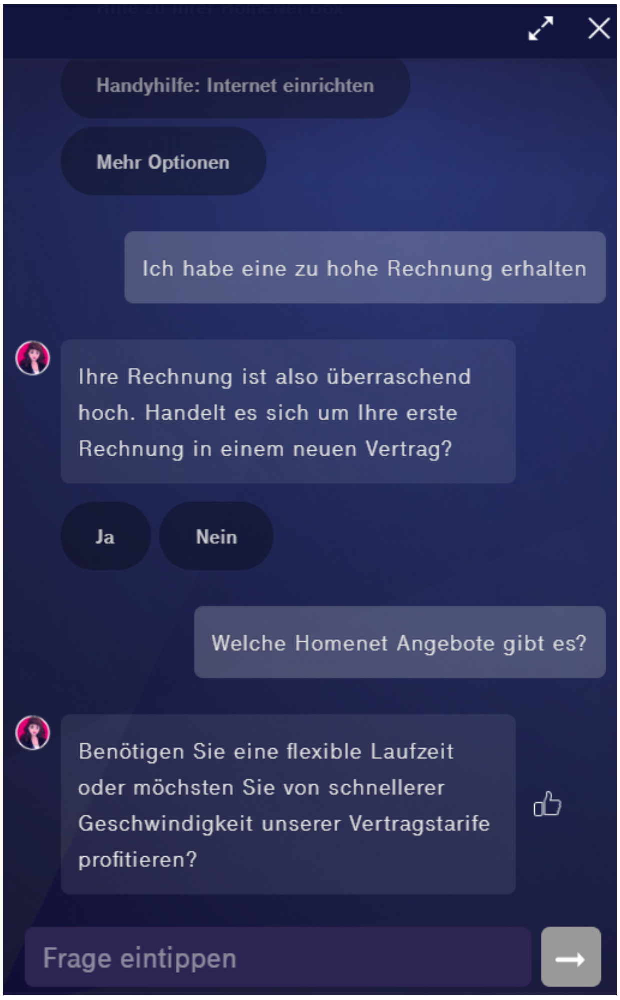

Figure 3.5: Chatbot conversation with multiple topics

Category	Number of Sessions
E-Mail	4127
Homenetbox	3705
Rechnung2	3302
Rechnung	3064
Rücksendung bei Defekt	2242
Sperren 2	2198
Roaming1	1947
HomeNetbox2	1498
Handyhilfe	1472
Sim-Karte - [...] Bestellung bis zum Tausch	1164
Sperren1	1088
PIN und PUK	1071
Zusatzpakete	899
Simlock für iPhones bei T-Mobile	864
Störung	800
Total	**34,651**

Table 3.4: Categories in chat sessions

3.1.2 Feedback Attributes

Is the user satisfied or not? Especially in the area of chatbots, explicit user feedback is essential for evaluation and further improvement of the chatbot. On the other hand, implicit feedback can be extracted out of the chat data by applying text mining techniques to measure users' preferences and satisfaction. As our dataset does not track users' actions after they leave the chat, the provided explicit feedback is the only source to measure user satisfaction directly. For feedback measurement and evaluation, our dataset provides three relevant attributes: *feedback label, feedback score, feedback comment*.

1. 'Feedback label' shows the scale of possible feedback scores given to the user
2. The 'feedback score' given by the user for a certain answer
3. 'Feedback comment' is a comment given in textual form

Feedback label & score

Having a closer look at the attributes feedback label and feedback score, our dataset X_{Tinka} captures two different feedback score ranges: First, chat sessions, where users have the possibility to give feedback by choosing one score out of {-2, -1, 0, 1, 2}. The smallest score of '-2' is equal to one star on the chatbot's interface and connotes dissatisfaction. The highest score is '2' and it represents a high satisfaction with the chatbot's answers. This feedback score is illustrated with five stars. The five-star scale is a Likert Scale, which was named after Rensis Likert [Lik32] who described this technique for assessments. The Likert scale is a five-point scale used to indicate how much the surveyed person agrees or disagrees, approves or disapproves to a given statement [AS07].

Secondly, a feedback scale where users can only choose one out of two scores: '-1' and '1' or 'like' and 'dislike' was used. Scales consisting of two answer possibilities are known as dichotomous scales [AS07]. In the exemplary chat in Figure 3.1 the answer is rated with '-1' as a downward thumb was selected.

The different feedback scales are attributable to the fact that the dataset X_{Tinka} captures chat data over seven months, from February to August 2016. During this period of time the cooperating telecom company changed the feedback options from a five-point scale to a two-point scale.

Out of the 673,176 interactions between the chatbot and users, in 636,111 interactions no feedback score is given. In 37,065 interactions users provide feedback by choosing a feedback score. 8,162 interactions are 'liked' by users and 26,344 'disliked'. Considering the five-stars feedback scale, 2,201 interactions receive only one star, 67 interactions get two stars, 41 three stars, 61 four stars and 189 interactions get a score of five stars. The Tables 3.5 and 3.6 summarize the feedback scores of the dataset X_{Tinka}.

Feedback	Number of Interactions	Share
liked	8,162	24%
disliked	26,344	76%

Table 3.5: Feedback scores (like-dislike scale)

Feedback	Number of Interactions	Share
one star	2,201	86%
two stars	67	2.6%
three stars	41	1.6%
four stars	61	2.4%
five stars	189	7.4%

Table 3.6: Feedback scores (five stars scale)

Two different feedback tables are displayed above due to the fact that the dataset X_{Tinka} captures chat data over several months. During this period of time the cooperating telecom company changed the feedback options from a five-point scale to a two-point scale.

Next, we analyze feedback scores according to the categories listed in Table 3.4. The top three categories, which occur the most in chat conversations ('E-Mail', 'Rechnung' and ‘Homenetbox’), are listed below in Table 3.7, 3.8 and 3.9 whereas the feedback for the categories ‘Rechnung’ and ‘Rechnung2’, as well as ‘Homenetbox’ and ‘Homenetbox2’ have been merged.

Among the top three categories, the highest amount of feedback is given in the category ‘E-Mail’ (761 interactions), followed by the category ‘Rechnung’ (509 interactions) and ‘Homenetbox’ (480 interactions).

In the category ‘E-Mail’ 668 feedback scores out of 761 (87.8%) are negative. A negative score is either a dislike button or the selection of one or two stars as a feedback. In the category ‘Rechnung’ 404 out of 509 interactions (79.4%) were rated negatively and in the category ‘Homenetbox’ 351 out of 480 interactions (73.1%).

Feedback	Number of Interactions	Share
liked	90	12.4%
disliked	636	87.6%
one star	32	91.4%
two stars	0	0%
three stars	0	0%
four stars	2	5.7%
five stars	1	2.9%

Table 3.7: Feedback scores in the category ‘E-Mail’

Feedback	Number of Interactions	Share
liked	86	19.2%
disliked	362	80.8%
one star	37	60.7%
two stars	5	8.2%
three stars	3	4.9%
four stars	8	13.1%
five stars	8	13.1%

Table 3.8: Feedback scores in the category ‘Rechnung’

Feedback	Number of Interactions	Share
liked	112	26%
disliked	319	74%
one star	32	65.3%
two stars	0	0%
three stars	2	4%
four stars	6	12.2%
five stars	9	18.4%

Table 3.9: Feedback scores in the category 'Homenetbox'

3.1.3 Attributes for Event Sequence Analysis

Interaction type

During a conversation between an user and the chatbot, different inputs come from the user's side as well as from chatbot's side. Each conversation input, also known as interaction, is assigned to an interaction type. The interactions can have one of these six different types:

- QA: Questions and answers by the users
- FAQClick: A FAQ link (in URL formate), which is provided by the chatbot and is clicked by the user
- LinkClick: A URL link, provided by the chatbot and clicked by the user
- Feedback: A feedback comment given by the user
- Dialog: Answers given by the chatbot (also includes some questions and answers by users)
- Output: Output by the chatbot

The interaction type 'QA' contains the biggest number of interactions: 291,370 (43.3%) and is followed by the interaction type 'FAQClick' with 125,160 (18.6%) entries. The 'LinkClick' interaction type has 118,464 interactions (17.6%), the interaction type 'Dialog' 101,116 (15%) and 'Feedback' has 37,065 entries (5.5%). A list of all interaction types and the number of interactions per type, can be found in Table 3.10.

Interaction type	Number of Interactions	Percentages
QA	291,370	43.3%
FAQClick	125,160	18.6%
LinkClick	118,464	17.6%
Dialog	101,116	15%
Feedback	37,065	5.5%

Table 3.10: Interaction types of chat sessions

Linkclick URL - Chatbot Answers

The attribute linkclick url saves the page URLs the users click. The chatbot has different possibilities to answer a user's question e.g. by providing a FAQ link, answering the question directly or redirecting to a customer service employee. One of these options is the link to a webpage where the user should find the answer to the proposed question.

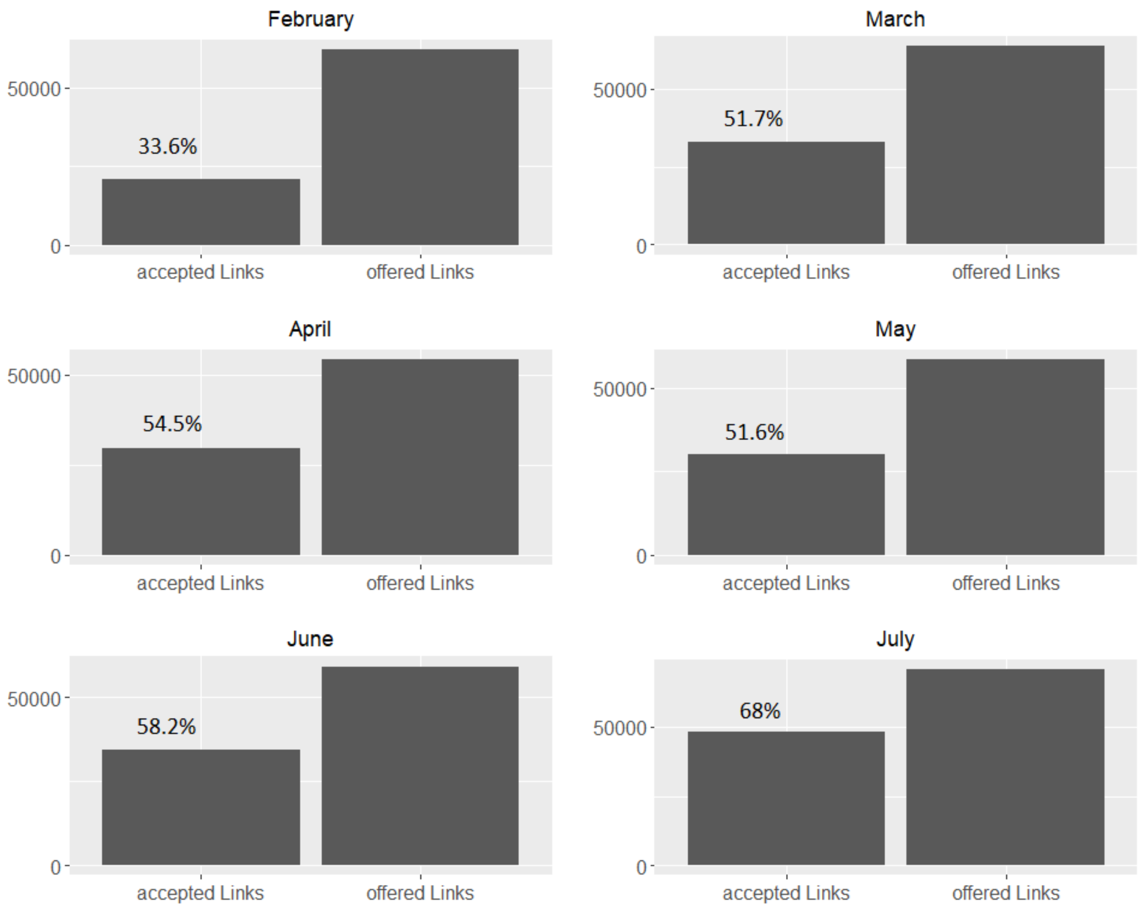

Figure 3.6: Comparison between offered and clicked links (Feb - July 2016)

In February, users clicked on 33.6% of the links the chatbot offered them. This rate of clicked links was equal to 51.7% in March, 54.5% in April, 51.6% in May, 58.2% in June, 68% in July and 74.4% in August, as it can be seen in Figure 3.6. Analyzing the evolution of these values, it can be assumed that over time the chatbot was improved in order to provide more accurate answers to users' questions. Although the share of clicked links increased from February till July to twice the original value, it is difficult to derive users' satisfaction out of this. As soon as the user clicks on the provided link, the chat conversation is interrupted and the user is forwarded to the clicked website. As no information on users' subsequent decisions is available, it is not possible to say if users are satisfied with the link which is provided as an answer. Nevertheless, it can be derived out of the presented values that the chatbot's answers lead to more websites been clicked in course of chatbot conversations. It might be also possible that users click on more answer link because the chatbot describes the provided links more accurately for example.

Previous Pages of Conversations

The previous page attribute captures the page a user visited before starting a conversation with the chatbot. Next, a list of these webpages can be seen in Table 3.11. The pages are extracted out of the dataset X_{Tinka} and ordered by their number of occurrences. Knowing the starting page of conversations, can help determining users' purpose for chatting e.g. a user, who starts the conversation at the billing page might have a question regarding his bill.

The FAQ page (*http://www.t-mobile.at/faq/*) is the starting page of 24% of all chat sessions. Almost a quarter of all users look up the FAQ page before asking the chatbot. The FAQ page is followed by the service website (*http://www.t-mobile.at/service/*) as the second most popular starting page. The service page gives an overview of all T-Mobile contact channels: the FAQ page, the livechat, Facebook, the service line, the community forum, Twitter, videos, etc. Furthermore, it is interesting to mention, that only 16% of all users start interacting with the chatbot right after the company's starting page (*http://mein.t-mobile.at/myTNT/start.page*) or the main page (*http://www.t-mobile.at/*). This gives evidence, that users first start searching on their own, before they ask the chatbot for assistance. Moreover, it is notable that five URLs include *[...]/myTNT/[...]*, which reveals that users starting their conversation at this link, were logged in to the system with their telephone number and personal credentials while chatting. In this case, background information of the user and user's contract details could have been utilized by the system to give personalized answers. A dataset, which merges all of this information, allows analyzing users in more detail and improves the chatbot's answer accuracy. Adding up the percentages of these conversations, it can be stated that 12% of all users were logged in during the chat conversation. Moreover, it is important to mention that some of the listed links below don't exist anymore as the website is developed continuously e.g. the page *http://www.t-mobile.at/angebot-bestandskunden-v/*.

Page	Occurrences	Share
FAQ page - http://www.t-mobile.at/faq/	50,621	24%
Service page - http://www.t-mobile.at/service/	34,632	16%
Start page - http://www.t-mobile.at/	23,242	11%
Start page for logged in users - www.t-mobile.at/myTNT/start.page	10,312	5%
Product page - http://mein.t-mobile.at/myTNT/product.page	4,791	2%
Login page - http://tgate.t-mobile.at/oauthlogin	4,208	2%
Invoice page - http://mein.t-mobile.at/myTNT/invoice.page	3,723	2%
Account page - http://mein.t-mobile.at/myTNT/account.page	3,362	2%
Online shop - http://shop.t-mobile.at/	2,612	1%
Service & Contact page - http://www.t-mobile.at/service/index.php	2,605	1%
Homenet - http://www.t-mobile.at/myhomenet/	2,347	1%
Online bill - http://mein.t-mobile.at/myTNT/portlet.page?shortcut=ebill	2,258	1%
Offers established customers - www.t-mobile.at/angebot-bestandskunden	2,050	1%
Favorites list - http://www.t-mobile.at/favoriten/	1,757	1%
Service packages - http://shop.t-mobile.at/tarife/voice	1,650	1%
Others	58,495	29%

Table 3.11: Starting websites of conversations

3.2 Data Preprocessing

Analyzing users' contributions in chat mediums, can lead to important information about the person chatting. Using these conversations, important social and semantic inferences can be made [ÖK10].

In order to apply algorithms on text extracted from chatbot sessions, first of all this data has to be preprocessed. First, we divide our dataset X_{Tinka} into sub-datasets based on the interaction type attribute assigned to the interactions. The following text analysis is particularly done based on the datasets of the interaction types 'QA', 'Dialog', 'Output' and 'Feedback' as these have textual data in contrast to the interaction types 'FAQLink' and 'LinkClick', which only contain URLs which are offered by the chatbot and clicked by the users.

Corpus Generation

Out of the previously described sub-datasets, three text corpora are generated: c_{QA}, c_{Dialog} and $c_{Feedback}$. Corpus c_{QA} contains primarily questions and answers by the user. Corpus c_{Dialog} includes answers of the chatbot and as the name already indicates corpus $c_{Feedback}$ is the feedback comments corpus. A corpus is described in the language science as a "body of written text or transcribed speech" [Ken14]. These corpora can be used for linguistic analysis and can lead to insights about the writer or speaker. Based on these corpora, the upcoming analyses are done e.g. the feedback analysis is based on the $c_{Feedback}$ corpus.

In order to generate corpora, first of all columns out of the dataset are selected, which include the textual data to be analyzed. These columns are summarized to vectors of characters and used as text sources for generating corpora. Next, the steps are described, which are conducted on the corpora during data preprocessing.

Eliminating Extra Whitespace

Extra whitespace is eliminated from the text.

Lower Case Transformation

All words are transformed to lower cases.

Stopwords Removal

Next, a list of words, which are known as stopwords, is removed from the corpus. Stopwords are words that usually don't increase the predictive capability of a text, e.g. articles like 'the' or pronouns like 'us' [WIZ15]. As our text is in German language a list of German stopwords is used for elimination. The words 'nicht' and 'nein' are kept although they are categorized as stopwords because an elimination of these negotiations leads to distortion of the feedback analysis.

Tokenizing

One of the initial steps of natural language processing is to identify tokens. Tokens are basic units, which can be extracted out of textual data. Therefore, the text has to be decomposed first. The most basic entity recognized as a token is a word [WK92].

The following table displays the number of tokens after different preprocessing steps:

	$c_{Feedback}$	c_{QA}, c_{Dialog}
Number of interactions	8,011	416,530
Number of tokens (token = word)	104,188	2,738,664
Number of unique tokens	78,894	2,090,237
Number of tokens after stopwords elimination	91,614	2,620,946
Number of unique tokens after stopwords elim.	68,683	2,007,304

Table 3.12: Overview token quantities

During the process of tokenization, the entire text from the dataset is decomposed in tokens. These tokens can be e.g. single words or sentences [MSB+14]. The chat data is tokenized into single words.

Stemming

Stemming refers to the process of converting tokens (as they have been defined previously) to a standard form. This step reduces the number of distinct types within a corpus as words such as 'types' and 'type' are counted as instances of the same type. During the stemming process words are grouped systematically and the number of unique words within a text is reduced [WIZ15].

3.3 Text Mining based on Chat Data

In contrast to classic data mining techniques, which are used to extract knowledge out of structured databases, text mining techniques have the aim to extract information out of unstructured textual data [RB98]. Using the following text mining approaches, we try to determine first of all users' information needs by analyzing frequently occurring terms and topics. As it has been mentioned previously, a subset of conversations is labelled with predefined topics by the data provider. The aim is to extract additional information on chat topics beyond these labelled conversations. Due to the huge amount of data available in textual format, extracting information out of collections of texts plays an important role in text mining. Extracting keywords out of data helps to learn the meaning of a text quickly as it saves going through all the text material manually [DPM+13].

After learning users' information needs, their feedback comments are analyzed in order to make statements about their satisfaction with the provided chatbot answers. The goal of feedback analysis is to discover users' opinions and satisfaction on a certain service, product or business [Gam04]. Gathering and analyzing user feedback is an essential step as it enables adapting services and products to maximize user satisfaction. User feedback can be categorized in explicit and implicit feedback depending on whether users share their opinion on the provided product or service intentionally or unintentionally. Feedback scores and comments as they are provided in our dataset, are examples for explicit feedback. Implicit feedback can be extracted by analyzing text data coming from users e.g. chat conversations [OTB+14].

Before knowledge can be extracted out of text, the raw text data has to be preprocessed. We generate corpora out of the conversations and apply stemming, tokenization and stopword removal on our text data as it is described in section 3.2. Next, the used text mining methods are introduced briefly and the gained results are presented:

3.3.1 Term-Document Matrices

Many data mining methods expect the input data to be in a highly structured format e.g. matrices. In order to easily work with textual data, which is usually represented as a collection of documents, it is recommended to transform textual data into a more structured layout [WIZ15].

For further work, we create term-document matrices out of the textual chat conversations. Term-document matrices are matrices, whose columns represent terms which occur within a corpus. Their rows correspond to a document of the corpus. The matrix displays the frequency of each single term that occurs within a specific document. Table 3.13 displays the term-document matrix of a small corpus consisting of two documents, 'DOC1' and 'DOC2' and three terms. The document 'DOC1' consists of the following sentence: 'T-Mobile is one of the biggest telecommunication companies of Austria and has thousands of customers'. Document 'DOC2' includes the following sentence: 'Telecommunication customers often face the information overload problem due to a huge amount of services and products offered in the telecommunication area.' The following matrix displays how often the three terms 'telecommunication', 'customers' and 'information' occur in the documents.

Documents	telecommunication	customers	information
DOC1	1	1	0
DOC2	2	1	1

Table 3.13: Sub-Matrix of a corpus containing 'DOC1' and 'DOC2'

In the term-document matrix generated out of the chat data, each interaction is one document of the corpus. Therefore the number of rows of the term-document matrix is equal to the number of interactions. The entries of the matrix count how often the words occur within an interaction.

3.3.2 Frequent Terms

Frequently occurring terms are extracted out of chat conversations easily based on the generated term-document matrices. The matrices give a good overview over the terms which occur in the conversations. Terms are extracted in order to determine the topics chat conversations are concerned with beyond the labelled categories. Taking a closer look at the terms, which occur most often in the corpora c_{QA} and c_{Dialog}, the information needs of users as well as the questions relevant to them can be understood. Instead of reading the entire conversation consisting of several words, having a look at the frequent terms is sufficient enough to discuss the topics of conversations .

The following Table 3.14 displays words, which occur frequently in the interaction type 'Question-Answer' and 'Dialog'.

Rechnung	Kuendigung	Internet
Handy	Verlängerung	Homenet
SMS	Flamingo	Email
Mobil	Vertrag	LTE
Telefonnummer	Tarif	Datenvolumen
Karte	Ausland	Box
SIM	Kuendigungsfrist	Net

Table 3.14: Frequent terms in corpora c_{QA} and c_{Dialog} (more than 3000 occurrences)

Out of the previously conducted frequent term extraction, it can be seen in Table 3.14 that certain terms occur often in chat sessions. In c_{QA} two main topics of terms become apparent. The first one is related to 'Vertrag & Rechnung' - billing and contract. Users ask questions about contract alteration e.g. extending or resigning the contract, about their bill, about using their mobile phone abroad and related questions. The second category of terms occurring often is related to the term 'Internet'. Chat users ask questions about their homenet, about surfing in the Internet, mail, router, the homenetbox, their data volume, etc. The optained results will be discussed in more detail in Chapter 5.

Next, the most often occurring terms per chat category are extracted. The categories are described in section 3.1. Table 3.15 lists the ten most often occurring terms per category. Based on the frequently occurring terms of the categories, we will next describe users' information needs. It is worth mentioning that the following descriptions are interpretations of the frequent terms in Table 3.15.

In category "Sperren" users ask questions regarding the locks of their sim cards and mobile phones. As the name already indicates, the category "Homenetbox" deals with questions regarding the Internet, the homenetbox and the router. Users of the "Email" category are looking for an email address to communicate with. Category "Rücksendung" deals with defects of users' mobile phones and how to repair/return/reclaim the devices. Conversations assigned to "Roaming" deal with questions concerning calling, messaging and Internet usage while being abroad. The terms "IPhone", "Galaxi" and "Samsung" occur in category "Handyhilfe". Conversations where users ask about sim lockouts, as well as forgotten pin and puk codes, can be found in the category "Pin & Puk". The additional service packages users ask about the most, are packages for foreign countries and additional Internet packages. In conversations of the category "Sim" users ask questions about sim activation, changing and time period extension.

3.3.3 Relationships between Words - Bigrams and Trigrams

Next, we analyze sets of two or three terms occurring together. The context words occur within is completely lost when the analysis is done at the level of single words. For example, the context of the phrase "mobile phone defect" is completely lost when each of the three words is only viewed individually. Moreover, it is interesting to analyze the relationship between different words, how often particular words occur together and what the predecessor or successor of a word is. A set of two successive words (w_1, w_2) is also called 'bigram'. The bigrams we analyse in context of this work are rigid collections. This means that the words w_1 and w_2 always occur in the same order and side by side without intervening words placed between them in the text documents. Analog a set of three words is named 'trigram' [DPM+13]. Bigram and trigram generation is realized based on the corpora c_{QA} and c_{Dialog}. Out of this chat text, each pair of bigrams is extracted and their occurrences counted. Table 3.16 represents the top 10 most often occurring bigrams in our chat text.

Sperren	Homenetbox	E-Mail	Rechnung	Rücksendung	Roaming	Handyhilfe	Pin & Puk	Zusatzpakete	Sim
handy	box	mail	rechnung	defekt	roam	iphone	sim	zusatzpakete	aktivieren
sperren	homenet	mobil	hoch	iphone	ausland	galaxi	brief	hallo	kart
entsperren	router	nein	warum	mobil	sms	gibt	find	sprachtarif	neuanmeldung
aufheben	nein	adress	mehr	partner	surfen	hallo	gesperrt	web	sim
lock	internet	email	nein	premium	telefonier	nicht	karten	zusatzdienst	anfordern
sim	unlimit	hallo	potentielle	shop	paket	samsung	kundenport	europa	neue
gesperrt	home	nicht	hallo	abgeben	roaming	hilf	puk	holiday	find
nummer	net	komm	rechnungen	einsendung	telefoni	handi	verloren	option	heraus
wurd	hallo	formular	nicht	onlineshop	vertrag	online	karte	datentarif	verlängerung
kart	wlan	find	einfach	onlinereparatur	zonenraum	shop	code	nicht	wechseln

Table 3.15: Most frequent words according to categories

First Word	Second Word	Number of Occurrences
kundenservice	kontaktieren	821
telefonie	sms	767
simlock	aufheben	722
thema	rechnung	531
einfach	erklärt	515
rechnung	einfach	514
email	adresse	490
wurde	gesperrt	441
los	geht's	404
online	shop	316

Table 3.16: Bigrams in chat data

The most often occurring bigrams are concerned with the topics sim card locks, billing, homenetbox and customer service. Reviewing the list of trigrams, in Table 3.17, it can be said that often occurring terms in the list of trigrams are sim card locking and unlocking, the homenetbox, customer service and roaming. Having an available list of bigrams and trigrams, it is also possible to determine in which coherence particular words occur.

Filtering the list of trigrams for the term "Internet" as the first word, we get the result that the trigram "internet funktioniert nicht" (equals "internet not working") occurs 238 times. The trigram "rechnung einfach erklärt" ("bill explained easily") is found 513 times in the analyzed corpus.

First Word	Second Word	Third Word	Number of Occurrences
sim	lock	aufheben	723
rechnung	einfach	erklärt	513
home	net	box	452
zonenroaming	für	vertragskunden	425
nummer	wurde	gesperrt	403
sim	karte	sperren	285
my	homenet	unlimited	276
neue	sim	karte	276
nein	kundenservice	kontaktieren	248
internet	funktioniert	nicht	238

Table 3.17: Trigrams in chat data

3.3.4 Chatbot Feedback - Correlations between Words

In this section feedback given to the chatbot in textual form is analyzed. Feedback in textual form is less structured than the feedback sources analyzed in section 3.1 but allows users to express their opinions independent from any given specification. By means of text mining techniques, the unstructured texts can be distilled down to main topics and recurring patterns of feedback [Gam04].

In sum our dataset includes 7,995 different feedback comments. Searching for words, which occur more than 1000 times within feedback comments, we receive a list of these words: ‘frage’, ‘nicht’, ‘antwort’, ‘hilfreich’. Words, which occur more than 500 times are ‘frage’, ‘wurd’, ‘nicht’, ‘antwort’, ‘hilfreich’, ‘verstanden’, ‘richtig’, ‘vorgeschlagen’. Next, the correlations between words is analyzed. The calculation of the correlation coefficient between words, can be illustrated by a short example of calculating the correlation between the two words ‘chatbot’ and ‘clickstream’.

	clickstream	*¬clickstream*	Total
chatbot	n_{11}	n_{10}	$n_{1.}$
¬chatbot	n_{01}	n_{00}	$n_{0.}$
Total	$n_{.1}$	$n_{.0}$	n

Table 3.18: Matrix correlation calculation between words

Table 3.18 displays how often the words ‘chatbot’ and ‘clickstream’ occur together within a document (n_{11}), how often ‘clickstream’ occurs without ‘chatbot’ (n_{01}), ‘chatbot’ without the word ‘clickstream’ (n_{10}) and the number of documents in which none of the two words occur (n_{00}). Using these values, the correlation between ‘clickstream’ and ‘chatbot’ can be calculated with the following *Pearson correlation coefficient* formula [BCHC09]:

$$\frac{(n_{11}n_{00} - n_{10}n_{01})}{\sqrt{n_{1.}n_{0.}n_{.0}n_{.1}}}$$

The Pearson correlation coefficient (PCC) is a widespread measure for correlation calculation and has a range between +1 and −1. A value of +1 represents perfect correlation and −1 perfect negative correlation [AP10].

As mentioned previously, the words ‘nicht’ and ‘nein’ are not removed during stopword elimination because an elimination of these negotiations leads to distortion of feedback analysis. Related words to the term ‘nicht’ are ‘hilfreich’ with a correlation of 0.50, ‘vorgeschlagen’ (0.48), ‘frage’ (0.43), ‘wurd’ (0.39), ‘verstanden’ (0.37) and ‘richtig’ (0.36). The word ‘antwort’ is correlated to ‘fehlerhaft’ (0.58), ‘frage’ (0.27) and ‘falsch’ (0.15). The correlation is calculated based on the occurrences of the words within the same document. As it is mentioned before, each document represents a single interaction.

Next, a wordcloud is generated displaying words, which users use most often while giving feedback. The resulting wordcloud is displayed in Figure 3.7. While analyzing the interaction type “feedback” we realize that once we have added the word “nicht”, which is a stopword and therefore removed while preprocessing the data, users’ feedback is very negative. Expressions such as “nicht hilfreich” are often found. Generating a word cloud out of the terms, which occur most often in feedback comments, we see a strong presence of the term 'nicht'.

Figure 3.7: Frequently used terms in the corpus $c_{Feedback}$

Next, we extract user feedback according to different chat categories as they are defined in Table 3.4. The analysis of user feedback according to categories allows us to make conclusions about the chatbot's answer accuracy according to certain categories. These categories, which are already described previously, are: "E-Mail", "Rechnung", "Sperren", "Rücksendung bei Defekt", "Roaming", "HomeNetbox", "Handyhilfe", "Sim-Karte - Alle Informationen von der Bestellung bis zum Tausch", "PIN und PUK", "Sperren1", "Zusatzpakete", "Simlock für iPhones bei T-Mobile" and "Störung". Table 3.19 contains the most frequent terms, which occur in users' feedback comments in each single category. As the availability of feedback data differs between the different categories, the last row displays the minimum number of occurrences of the top three feedback terms in the table. In order to get three words, which occur most often in a category, the parameters of occurrence have to be scaled down to a very small number. In the category 'Handyhilfe' these three words occur at least six times. When this number is increased to ten, then less than three words are displayed. Having this kind of infrequent dataset, it is difficult to make statements about particular categories and user feedback comments.

Sperren	Homenetbox	E-Mail	Rechnung	Rücksendung	Roaming	Handyhilfe	Pin & Puk	Zusatzpakete	Sim
nicht	frage	email	potentielle	hilfreich	ausland	frage	frage	nicht	frage
sim	nicht	find	rechnung	nicht	Roaming	nicht	nicht	frage	nicht
sperren	internet	frage	Rechnung	vorgeschlagen	antwort	vorgeschlagen	verstanden	find	verstanden
12	12	17	12	20	7	3	10	6	11

Table 3.19: Most frequent words in users' feedback comments according to categories

Summarizing the Results

In the preceding section we demonstrate one possibility of determining information needs and preferences by using text mining on the available chat data. The extracted needs, satisfaction and preferences of users can be summarized as follows.

Using chat categorization, frequent term extraction, as well as bi- and trigram extraction, the following top topics become apparent:

- Billing: One of the most popular topics of chat conversations is billing. The category “billing” has the biggest share of 6,575 (19%) conversations compared to the remaining 28,285 chat sessions (see Table 3.4). Users contact the chatbot because they have questions regarding their recent bills. The trigram “rechnung einfach erklärt” has the second highest proportion in the list of trigrams in Table 3.17. Users asking questions about bills are obviously customers of the telecommunication company. To increase user satisfaction, it would be important that the chatbot could see into the bill of the specific user and give personalized answers. As the chatbot cannot give personalized answers even if the user is logged in to the system, it is forced to forward the users to customer service employees.
- Homenet: Homenet is the Internet service the telecom provider offers. 5,206 sessions (15%) are counted as chats belonging to the topic “Homenetbox” as it can be seen in Table 3.4. The term “homenet” also occurs in the list of most frequently occuring words in chat conversations. Most of the users contact the chatbot due to issues with their Internet router or because their Internet is not working.
- Locking and unlocking simcards: The trigram “sim lock aufheben” (“pick up sim lock”), which has the highest proportion compared to all trigrams, occurs 723 times in conversations (see Table 3.17). “Simlock aufheben” and “wurde gesperrt” are bigrams, which can be found in the top ten of all bigrams within the chat data. Moreover, the category “Sperren” counts 3,286 (9.5%) out of 34,651 chat sessions. In these conversations telecommunication users ask the chatbot for help because their mobile phones have been locked and they are not able to use them anymore. Moreover, there are also users who have forgotten their pin or puk code and therefore contact the chatbot (3%).

Knowing users’ topics of interests enables understanding their information needs. Furthermore, knowledge of users’ needs can be used to increase their satisfaction by helping them to find the right answer quickly and easily. As it is demonstrated in Table 3.3, 122,540 out of 215,859 chat sessions end within one minute. This testifies that users are not willing to spend a huge amount of time chatting with the chatbot and want the solution to their problems as soon as possible. In context of this condition, the knowledge of popular topics, questions and problems can be very beneficial. The chatbot could offer these topics right at the beginning of the conversation as options to the users.

As it is mentioned before, the available data captures conversations from February to August 2016. Comparing the chatbot from 2016 to today's chatbot (January 2019), improvements resulting from the knowledge of popular questions and topics can be observed. Immediately after starting the chat, the user has the possibility to choose one out of the following options: "answers around the topic billing", "assistance for the homenetbox", "support for mobile devices: installing the Internet" and "more options". These options correspond to the information needs extracted out of conversations in this thesis. More personalization can be integrated into the chatbot, if it asks users to log in with their phone number. Personal information, as well as their past topics of interests can be used for recommendation.

Chatbot feedback is an inevitable source for determining users' satisfaction. Resuming the analysis of users' feedback, it can be said that feedback given to the chatbot is mainly negative (77%) and users are not satisfied with the chatbot's answers. In Figure 3.12 users' feedback scores are summarized. Moreover, it is valuable to analyze feedback according to the category it was given for and to compare the categories to each other. Analyzing feedback of the top three categories ("homenetbox", "E-mail" and "Rechnung"), it becomes apparent that feedback provided to chats dealing with the topic e-mail is more negative (87.8%) than for homenetbox (73.1%) and billing (79.4%).

The obtained results of the previous section are discussed to a greater extend in chapter 5.

3.4 Event Sequence Analysis

As chatbots are an emerging topic in the area of recommender systems, the interpretation and evaluation of chatsbots is still in its early stage. A first step towards this goal, is to transform the chat conversation data into a more structured form. Event sequence analysis is a method for analyzing chats beyond the widely used method of text mining. Event sequence analysis allows focusing on the chat sequence as a whole rather than analyzing only individual questions, answers and feedback comments as it is done using text mining methods. As it has been mentioned before, each interaction of a chatbot conversation has a certain interaction type. In total, five different interaction types exist for our chat conversations in the dataset: QA, FAQClick, LinkClick, Feedback and Dialog. Each interaction in the dataset provided by T-Mobile has one of these five interaction types assigned. This categorization of interactions allows us to analyze conversations beyond text mining methods and discuss the structure of conversations as sequences of events.

It has been communicated by the provider of the chat data, T-Mobile Austria, that the chatbot gives answers to users based on predesigned chat conversations. These conversations have been designed prior to the development of the chatbot and are based on the experiences of T-Mobile customer service employees. Depending on the questions asked by users, the rule based answering system behind the chatbot determines to which of the designed conversations it can align the question in order to deliver an answer. Whether or not the designed conversations align with real user behavior, has not been evaluated at the present state. Modelling chat conversations as networks of event sequences provides the possibility to analyze these predefined chat conversations based on real user data and behavior as it is demonstrated in section 3.4.3.

3.4.1 Introduction to Network Analysis

Aggregating the event sequences of different chat sessions, a network of events can be obtained. Before analyzing the generated event networks in detail, we first give an introduction to basic concepts of network analysis:

In the mathematical context a *network* is also known as a *graph*. A graph $G = (V, E)$ is composed of a finite set of nodes V, also known as *vertices* and a finite set E containing the *edges*. Nodes are fundamental units of graphs and are connected to each other by edges [New18].

Graphs can be either *directed* or *undirected*, according to their edges. Directed edges are usually represented in visualized networks as arrows. In undirected graphs, edges are unordered pairs of two nodes. Whereas each edge of a directed graph has a direction, leading from one node to another one. Furthermore a *weight* can be assigned to the edges of a graph. Depending on the use case, weights can be assigned to edges for different usages, representing e.g. the length, capacity or costs of an edge [New18].

In an undirected graph the *degree* of a node $deg(n)$ signifies the number of nodes, which are connected to node n. They are also called neighbors of n. The degrees in a directed graph are divided in *in-degrees* and *out-degrees*. The in-degree of a node is the number of edges pointing inward to the node. The out-degree is the number of nodes pointing outward from it. Taking webpages as an example, the in-degree of an Internet site X is the number of sites linking to X and the out-degree is the number of pages to which page X links [New18].

Introduced by Page et. al [PBMW99], the *pageRank* of a node measures the importance of the node within the network. It is assumed that the importance of a node is related to the number and pageRank of other nodes which link to it. A simplified version of the pageRank algorithm can be calculated as follows for node u.

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v}$$

B_u is the set of nodes linking to u, $N_v = |F_v|$ the number of links from node v, which preceded u, and the factor c is introduced to normalize the calculated value.

3.4.2 Data Preprocessing

To model chat data as event sequences, the first step is to extract events out of the conversations in dataset X_{Tinka}. While preparing the data, based on the interaction types, for each interaction of a session, a label is extracted out of the existing dataset. For the interaction type “QA” labels such as “qa_745” are already included in the dataset. The QA-labels start with the sub-string “qa_” and are followed by a number, which marks a certain question. When a user in another chat session asks the same question, this question is also labeled with “qa_745”. The entries of the interaction type “Dialog” are labeled by shortcuts starting with “dn-” and end with a number. The “FAQClick” is categorized by “faq-” and also followed by a certain number tagging the FAQ link. In case of interactions of the type “LinkClick”, the URL for the link is simply extracted. Users’ feedback is extracted by constructing a label out of the term “Feedback” and the feedback score given. The feedback scores range from -1 to 2, e.g. “Feedback_1”.

In Figure 3.8 an exemplary conversation and events can be seen, which are determined within that conversation. The first event of this conversation is the propose of a question by the user: “Wie lange dauert die Aktivierung bei einem neuen Vertrag?”. This question is categorized with the tag “qa_745”. This categorization is very useful as it helps to cluster conversations based on the topics they refer to.

The second event is the answer, which is given by the chatbot. This answer contains three links: 1. a link leading to T-Mobile’s onlineshop, 2. a link for the website *mein.t-mobile.at* and 3. a link for the customer service. If the user would have chosen one of these three links, this would have been the second event of our event sequence. In our conversation the user does not select a link, but terminates the conversation by giving feedback (label: “Feedback_2”), which is the second and last event of the sequence.

These events are extracted for each single chat session and summarized within one dataset: X_{events}. In sum, the dataset contains 3,181 unique events. The events can be questions, answers, links to FAQs, links to websites and feedback events. The first column of our dataset X_{events} contains the sessionID. Based on this attribute it is possible to determine the sequences of events for each single chat session. The second column is the first event of each conversation, the third column equals the second event, etc.

Based on this set of chat events, it is possible to analyze the conversations e.g. the length of conversations, which questions are very popular, if answers are useful or not and what the ending events of conversations are. A detailed analysis of the events and their sessions is presented in the following section.

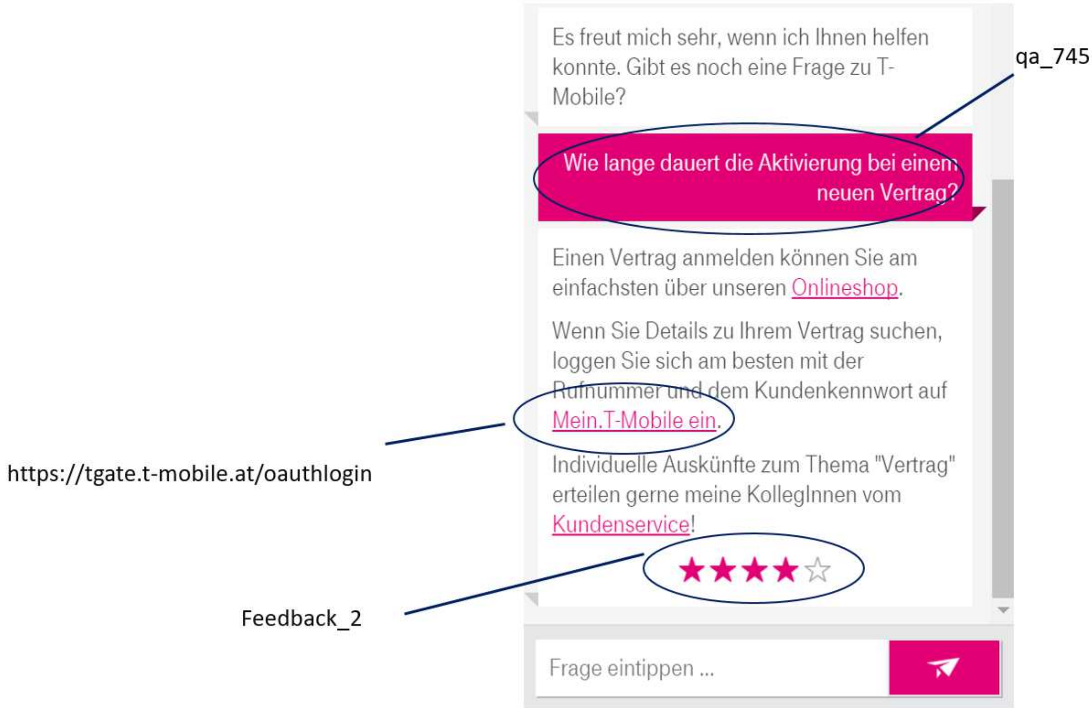

Figure 3.8: Labelling of chat interactions

3.4.3 Chat Conversations as Networks of Events

In Figure 3.9 a subset of the extracted events of February are presented as a network. Generating a network graphic based on the dataset X_{Tinka} was not possible due to the huge number of nodes and the available computational power.

Converting all chat sessions of February to sessions of events and presenting them within one network, we have in total **3,181** nodes and **917,071** edges.

In our directed network, each node represents one event. Two nodes are connected, if the two events happened successively with the direction of the edge pointing from the first event to the second one. Weights assigned to the edges indicate how often two events take place consecutively. The sizes of the nodes represent their *pageRank* score, as it is described before. Question nodes have labels starting with "qn-" and all answer node labels start with "dn-". Question and answer nodes are colored black in our networks. The red nodes are FAQ nodes and link clicks. As it can be seen in figure 3.9 two nodes are colored orange, which are the feedback nodes. The huge orange node represents negative feedback scores and the smaller orange node positive feedback.

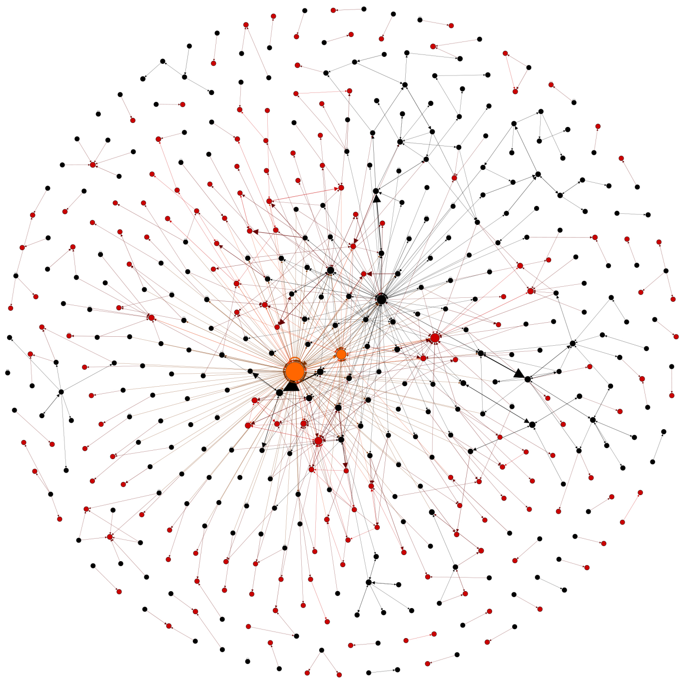

Figure 3.9: Sub-network of events in February 2016

As it has been mentioned before, chatbot's answers are generated by a rule based system which matches users' questions to predefined conversations. According to the data provider the matching of a conversation to a question is interpreted as successful if users don't interact further with the chatbot after receiving an answer. In order to evaluate the predefined chat conversations against real user behavior, a measure is introduced to determine if users continue to interact with the chatbot after getting a certain answer.

In the system the weighted answer nodes are marked by specific labels starting with “dn-”. Thus, nodes that have no or a small amount of edges leaving the node, are supposed to have a low weighted out-degree $outd_w$ compared to their weighted in-degree ind_w. On the other hand, nodes representing answers that have a high amount of interactions following the answer, have a high weighted out-degree compared to the weighted in-degree. Based on this idea, we propose the following measure q_n to evaluate the degree of interactions occurring after the chatbot gave an answer:

$$q_n = ind_w/(ind_w + outd_w) = ind_w/d_w$$

Thus, nodes with q_n close to 1 represent answers after which many users stopped interacting with the chatbot, nodes with a much lower q_n represent answers that typically lead to an ongoing interaction with the users. A score of 0.5 indicates that all users who reached the specific node, continued their search afterwards.

Figure 3.10 displays a boxplot generated out of the q_n values of conversations from the month February. The minimum value of the boxplot is equal to 0.5. In this case the ingoing degree is equal to the outgoing degree and no chat conversation ends after this node. Whenever the chatbot provided this answer in a conversation, the conversation continued afterwards. The maximum value of the boxplot is equal to one. Nodes having an answer quality of one, don’t have any outgoing edge as no user continued the chat conversation after reaching this node. The boxplot is right skewed and has a median (0.625) closer to the first quantile (0.515) than to the third quantile (0.793). Table 3.20 summarizes the answer quality by stating the minimum value, the maximum value, the quantiles and the median. The minimum is equal to 0.5 and the maximum to one. The 0.25, 0.5 and 0.75 quantiles are located at 0.515, 0.625 and 0.793.

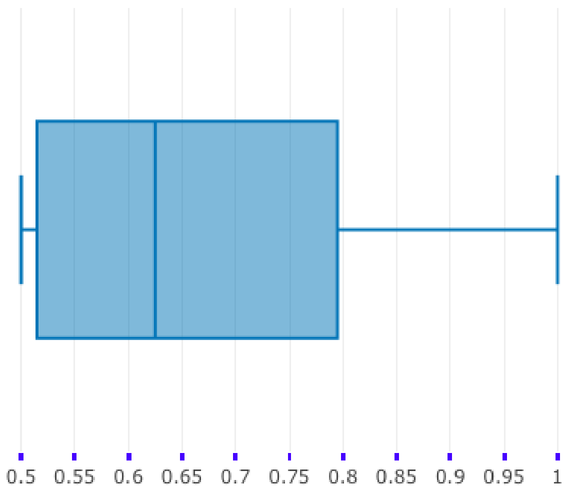

Figure 3.10: Boxplot of answer qualities in February 2016

Min	1st Qu.	Median	Mean	3rd Qu.	Max
0.5	0.515	0.625	0.686	0.793	1

Table 3.20: Summary of q_n values in February

Figure 3.11 splits and presents the answer quality in more detail. As the histogram displays, the highest number of recommendations has a quality rate around 0.5, followed by one and other values smaller than 0.7.

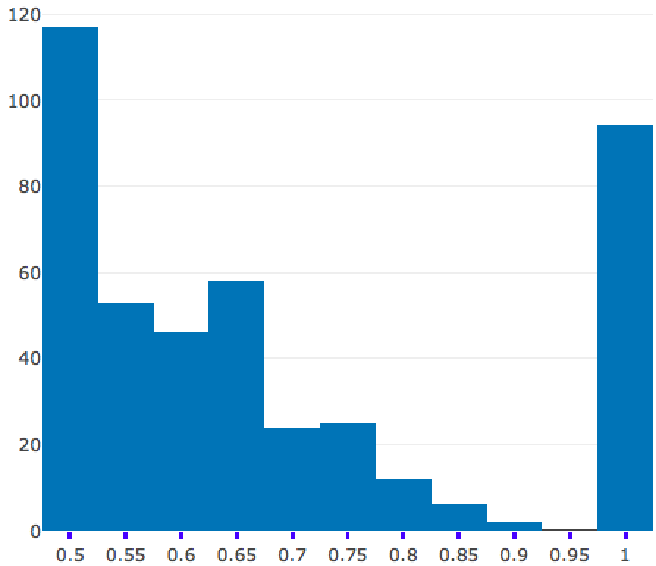

Figure 3.11: Histogram of answer nodes according to q_n value (Feb 2016)

In Table 3.21 the five nodes of type LinkClick with the highest recommendation score and the five nodes with the lowest recommendation score for the month of February are displayed. We select nodes with a weighted degree higher than 10 in order to ensure that no generalization is defined due to to the reason that a link only occurs in a single conversation.

Analyzing the q_n value of answer nodes, enables us to identify inappropriate implementations of answer nodes. If a chatbot answer has a answer quality value of 0.5, this implies that whenever this answer was given, the user continued with the search process. The answer might not fit to users' questions.

Answer node (type LinkClick)	ind_w	$outd_w$	q_n
Phonebook - https://mein.t-mobile.at/myTApp/phonebook	12	0	1.000
FAQ link - http://faq.t-mobile.at/app/answers/detail/id/470/	57	3	0.950
FAQ link - http://faq.t-mobile.at/app/answers/detail/id/3897	17	1	0.944
Apple watch - https://www.t-mobile.at/apple-watch	14	1	0.933
FAQ link - http://faq.t-mobile.at/app/answers/detail/id/3923	301	27	0.918
Additional packages - http://www.t-mobile.at/zusatzpakete	13	10	0.565
Invoice - https://www.t-mobile.at/erechnung	14	11	0.56
Starter pack - https://www.t-mobile.at/basispaket	57	47	0.548
Notice periode - http://www.t-mobile.at/kuendigungsfrist	128	112	0.533
Direct debiting - https://www.t-mobile.at/bankeinzug	21	21	0.500

Table 3.21: Highest and smallest q_n values of links

Summarizing Results

Event sequence analysis has been chosen as an appropriate approach for evaluating the predefined chat conversations based on which the chatbot answers users' questions. The main results are listed below:

- Half of the answer nodes, which include links and answers given by the chatbot, have a small q_n value between 0.5 and 0.625. This gives evidence that the majority of users who received these answers, continued chatting with the chatbot afterwards. As users' questions are matched to predefined conversations and the aim is that these answer nodes are ending points of conversations, the q_n values help evaluating these answers and determine the conversations which have to be adjusted.

- The node with the highest value of 1 is *https://mein.t-mobile.at/ myTApp/phonebook.html* where users can find their personal online phonebook. In twelve conversations users get this event as an answer to their question and in none of these twelve conversations the user continues to chat with the chatbot afterwards. The recommended website allows users to configure their telephone numbers in the phonebook. It can be assumed that the answer quality of this answer node is high because probably there is no possibility of confusion with other questions. In contrast to questions regarding the billing for example, where the user can ask many different questions. The nodes faq_140 *(http://faq.t-mobile.at/app/answers/detail/a_id/470/)* and faq_3897 *(http://faq.t-mobile.at/app/answers/detail/a_id/3897)* have a high answer quality of 0.95. Out of 57 users, only 3 users continue the chat conversation after receiving faq_140 as an answer and only one out of 17 users continues asking after getting to node faq_3897. As the website is continuously improved and developed further, the links to both FAQ questions are not available anymore. Therefore it is not possible to determine the topics of these websites.

- Having a look at the answer nodes with the smallest results is even more interesting as this knowledge helps to identify and eliminate falsely installed answer nodes. Each of the 21 users who received the link *https://www.t-mobile.at/?etquestion=bankeinzug* as an answer, continued to chat. This fact implies that this answer node has to be reconsidered and if necessary, exchanged. Further nodes with a small answer quality are (1) notice periode - *http://www.t-mobile.at/?etquestion =Kuendigungsfrist*, (2) basic service packages - *https://www.t-mobile.at/?etquestion=basispaket*, (3) online bill - *https:// www.t-mobile.at/?etquestion=erechnung* and (4) additional service packages - *http://www.t-mobile.at/handytarife/ zusatzpakete/SMS_MMS*. In order to improve users' satisfaction, conversations which include nodes with small answer qualities have to be analyzed in more detail. The falsely implemented answers have to be reconsidered and adjusted.

Finally, it is important to mention the limitations we faced while analyzing the event sequences of chat conversations. First, no information on user actions after chatting is available. In order to determine an answer's quality holistically, analyzing user behavior after the chat is necessary. This is also the key learning of the conducted event sequence analysis. A dataset is needed which connects users' chat conversations to further data sources e.g. their click behavior on the website after chatting or their conversations with customer service employees afterwards.

3.5 Evaluation & Conclusion

In this chapter users' information needs, preferences and satisfaction are analyzed based on chat conversations between users and chatbots. The proposed methods and models are implemented based on available data of a telecommunicaton chatbot. The chatbot answers questions regarding services and products of a telecommunication company.

One main criteria for choosing the applied techniques and models, was the general applicability of the approaches. The work conducted in scope of this thesis, was done based on data provided by an Austrian telecommunication provider. The focus was put on the development of generic models as the conducted work should be applicable for other chatbots and datasets as well.

The applied approaches and techniques, as well as the generated models are evaluated with the help of domain experts. Valuable insights were provided to the data provider on the one side, and their aid and knowledge was used for choosing the best approaches on the other side. First meetings with the data provider took place to discuss the provided dataset. These meetings were also used to discuss limitations and questions regarding the understanding of the dataset. Next, the results of descriptive analysis were presented and further proceedings were discussed. During these discussions, it could be found out that there existed the need of analyzing chat conversations beyond text mining as this was partly done by the telecommunication company itself in the past. Event sequence analysis was presented as a sufficient method to reach this goal. After the results were obtained, a final meeting was held to present and discuss the outcomes of the analyses. The generated models and gained insights were very strongly practicable and could provide meaningful insights into the needs, preferences and satisfaction of the user chatting with the chatbot.

At the beginning of the chapter an overview of the available dataset is provided. Table 3.1 categorizes the attributes used for analysis in three groups: basic attributes, feedback attributes and event sequence analysis attributes. Basic attributes (interaction ID, session ID, timestamp, user ID, dialogname) provide an overview of the available data. For feedback logging the dataset has three different attributes: feedback label, feedback score and feedback comment. Text mining is done based on the attributes interaction value and feedback comment. The dataset for event sequence analysis is created based on three attributes of the initial dataset, namely previous pages, interaction type and linkclick url.

One research question, as well as three sub-questions, are stated at the beginning of this chapter. First, the results of the sub-questions are presented and finally in chapter 5 the main research question is discussed.

> **RQ1.a** "Which user interaction attributes are relevant and determinative for fulfilling this purpose?" (i.e. understanding users' information needs and preferences)

For answering this question, the following attributes are determined as relevant and, subsequently, will be discussed in more detail:

- The starting page of users' interactions
- The time when each interaction occurs
- A categorization of chat conversations according to products/services/topics, etc.
- Feedback in form of a dichotomous scale
- Feedback in textual form
- User identification to observe user behavior over different sessions
- Attributes capturing user behavior after the chat conversations

Tracking the start and ending time of conversations, allows us to make useful statements about the workload of the chatbot at different times. *When do users prefer to chat with the chatbot? At which day of the week or at which time of the day do most of the interactions take place?* Chatbots, which are frequently used by a huge number of users, need this information to manage their workload and capacities. This is done in section 3.1.1 for the Tinka chatbot. Figure 3.2 presents a peak of conversations between 8 and 10 o'clock. The most preferred days of conversations are Monday and Tuesday, followed by Wednesday, Thursday, Friday, Saturday and Sunday. More than half of all conversations between February and August 2016 took place between Mondays and Wednesdays. The majority of users like to chat with the telecommunication chatbot during the office hours and on weekdays. It is assumed that this is due to the reason that during these time frames talking to a customer service employee is more difficult.

For understanding users entirely with their needs, preferences and characteristics, it is important to observe users over a period of time. Based on the available dataset, user behavior can be analyzed only during single sessions. In Figure 3.4 the duration of sessions between February and August 2016 are presented. During these timespans the observation of users' chat behavior is possible. More meaningful insights about users can be gained if it would be possible to study users across different chat sessions e.g. if a user returns to the chatbot and asks for the same question or a similar question again. In this case the chatbot would not give the same answer to every user but answer individualized to users' histories of previous chats. As mentioned in section 3.1.1, the user id is not provided in our dataset and therefore tracking and analyzing users across sessions is not possible. One important lesson learned out of the analyses, is the importance of user ids. While answering the research question, which user interaction attributes are relevant and determinative for understanding users, an attribute for user identification cannot be omitted.

Attributes capturing user actions before and after chat conversations are valuable and important as well. Otherwise it is not possible to determine users' reasons for leaving a chat session. On the one side a user can stop chatting because the right answer was given by the chatbot and the user is satisfied. On the other side, a conversation can be terminated because the user is desperate and convinced that the chatbot cannot give the right answer. Concerning other use cases e.g. an online shop assistant chatbot, it is important to know if users buy the proposed articles after the chat to evaluate the chatbot. Furthermore, knowing users' behavior before a conversation is useful to give appropriate answers right at the beginning of a conversation and this can help to overcome the cold start problem which was mentioned previously.

Within the context of this thesis, the analyzed chat data is labeled according to the topic of the conversation. Figure 3.5 shows the most often occuring topics of conversations in the entire dataset X_{Tinka}. Analyzing frequently occurring terms within the context they occur, can provide substantially more information about users' needs and preferences. For example, the term 'nummer' does not give any reference to the context it occurs in, but the trigram 'nummer wurde gesperrt', in contrast, reveals that the sim of the user is locked and that the user is looking for a solution regarding this problem. The following topics have been extracted and determined as important based on their frequency: billing, homenetbox, simcard locking and unlocking, as well as customer service.

RQ1.b "How can explicit, textual feedback and inexplicit information extracted from the chat data be utilized in order to determine users' satisfaction?"

Having no information about users' activities after the chat, chatbot feedback becomes an important source of measuring satisfaction. The available dataset captures feedback in terms of scores and comments. Figure 3.12 displays a summary of feedback scores given to the analyzed chatbot. Resuming the analysis of users' feedback, it can be said that feedback given to the chatbot is mainly negative (77%) and users are not satisfied with the chatbot's answers. Users have the possibility to rate the chatbot using the provided scales. The Likert scale, which the telecommunication company provided first, gave users the option of rating the chatbot on a scale from one to five stars. Later, the scale was adapted and changed to a dichotomous scale providing two options for feedback: an upward thumb (like) and a downward thumb (dislike). One star, two stars and a click on the dislike-button are interpreted as negative feedback. Whereas five and four stars, as well as clicking on the like-button is understood as positive feedback. Nevertheless it is important to mention, that human beings tend to give feedback more often when the feedback is negative than giving positive feedback. Figure 3.12 summarizes and visualizes the feedback ratings. Out of 673,176 interactions, in 94.5% no feedback score is provided. The remaining 5.5% can be divided in 23% positive feedback and 77% negative feedback.

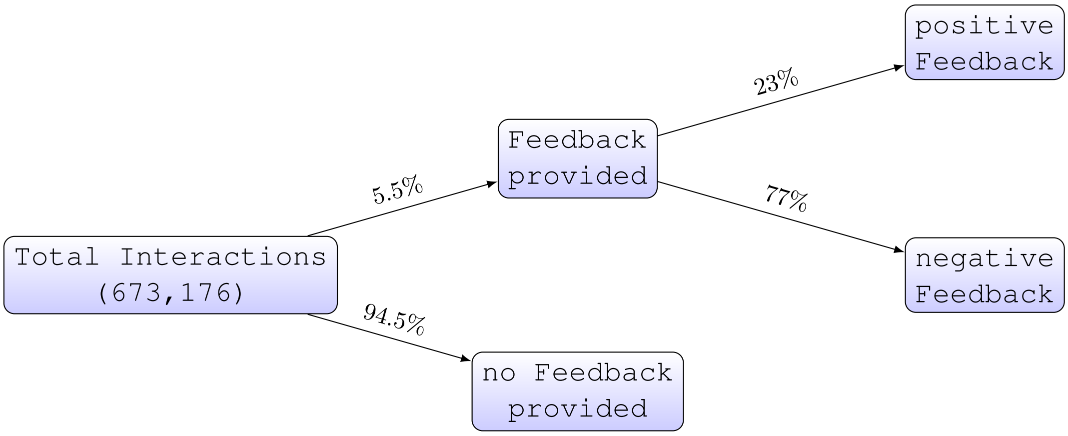

Figure 3.12: Summary of feedback scores

As it has already been mentioned in section 3.1.2, a short feedback scale should be chosen as this forces users to take a position whether they are satisfied or not. Answer ratings of three stars can neither be counted as positive feedback, nor as negative feedback. The knowledge of rated answers e.g. accoring to categories can be utilized to improve the chatbot specifically for the given category. In Table 3.7, 3.8 and 3.9 feedback scores of the most often occurring categories are displayed. Section 3.1.2 points out that feedback distinguishes depending on the category it is given for. Analyzing feedback for the top three categories ("homenetbox", "E-mail" and "billing"), it becomes apparent that feedback provided to chats dealing with the topic e-mail is more negative (87.8%) than for homenetbox (73.1%) and billing (79.4%). Such findings can help to improve the chatbot's answers in negatively rated categories.

Feedback in form of text can be very noisy. Analyzing feedback in textual form is still important as it allows users to express their opinions independent from any given structure e.g. scores or surveys [Gam04]. Gaining insights of users' satisfaction by analyzing their comments has been more challenging than expected. The majority of feedback comments consist of statements such as 'the answer was wrong.' or 'the answer did not help me.' and vulgar language is often used to express one's dissatisfaction. Users prefer to give feedback scores rather than writing that the answer was dissatisfying. Moreover, further information on reasons for dissatisfaction or why the given answer did not match is not provided. Feedback comments were also extracted according to chat topics, which was not very informative due to the small amount of labelled feedback comments (see Table 3.19). Comparing the feedback scores, as they are summarized in Figure 3.12, to feedback comments it can be said that more information on satisfaction can be extracted based on the given scores. Although only for 5.5% of interactions a feedback score is given, comparing the negative and positive feedback scores to each other is valuable to determine the topics which have the highest amount of positive or negative feedback (see Tables 3.7, 3.8 and 3.9).

> **RQ1.c** "How can non-textual information gained from the chat data be used for determining whether users' information needs were satisfied or not?"

Event Sequence Analysis has been chosen as an appropriate method to answer this question. The aim is to organize chat data in a more structured representation in order to analyze chat data beyond text mining methods. Chat conversations are understood as sequences of interactions. Each sequence has a previous page, which is the Internet site where the conversation starts and a sequential list of interactions. Analyzing the starting page of conversations, results that approximately one fourth of all users (24%) first looks at the FAQ page before starting a chat. These users evidently do not find the right answer on the FAQ page and start a conversation with the chatbot.

Event sequence analysis has been proven to be a good approach to evaluate the chatbot's answers. Calculating the self-defined q_n measure helps to evaluate the matching of questions to predefined chatbot answers. Figure 3.10 and table 3.20 present these values for chatbot's answers in February 2016. As it can be seen, half of all answers have a small value between 0.5 and 0.625 compared to the entire scale (0.5-1). The node with the lowest answer quality is *https://www.t-mobile.at/?etquestion=bankeinzug* which answers users' questions about automatic debit transfer. Users received this answer node in 21 conversations and in each conversation the user continued the chat conversation. The nodes with the highest values are also listed in Table 3.21.

Finally, with regard to the stated research question, event sequence analysis is a sufficient method for evaluating the predefined, rule based conversations and their answers but it is not possible to determine if a user is satisfied after receiving an answer or not. We know when and after which answer node conversations end but no information is available if users left the conversation out of frustration or because they are satisfied with the answers given. In order to determine users' satisfaction, we have to rely on the feedback analysis results described for research question RQ1.b.

CHAPTER 4

Clickstream Analysis

The following chapter is devoted to the topic of clickstream analysis. A clickstream is a sequence of HTTP requests a user sends to a certain website. Tracking users' website behavior facilitates finding users' topics of interests, how long they search for something, what similarities they have to other users, etc.

In this chapter we aim to answer the following research question: "How can structured, non-textual website clickstreams be used to predict users' future actions?" (RQ2).

In contrast to chatbot data, which is only available for those who have installed a chatbot on the own website, clickstream data is widely available. Everyone offering an online service, can log users' click behavior on the website. Moreover, the datasets resulting from these logs do not strongly distinguish from one another. Each dataset will contain, at the minimum, websites' URLs and a key indicating which clicks belong to the same session.

First, this chapter describes the available dataset and its structure. The different data preprocessing steps, which are performed to prepare the dataset for clickstream analysis are presented in section 4.2. In order to answer the research question, a Markov chain model is developed and trained. Predictions are generated using this model and finally, in chapter **??**, the analysis is evaluated and concluded.

4.1 Overview and Descriptive Analysis

The clickstream dataset used for analysis in this chapter, is provided by T-Mobile, the cooperating telecommunication company. The data is stored in five textfiles, whose sizes range between 2,674 KB and 9,530 KB. The provided files capture clickstream sessions from the 27th March, 2017. The clickstreams track users' clicks during their interactions with the company's homepage. The homepage of the telecommunication company can be accessed using the link *www.t-mobile.at*. Figure 4.1 displays the starting page of the analyzed website. On the far left side of the page the menu with links to access the product tab, the service tap, the customer tap and the login can be found. Moreover, the homepage contains an icon for chatting with the company's chatbot and another icon to start a chat with a real customer service employee. Over the product tab one can access different kinds of smartphones, service packages, smart home devices, internet packages and combined offers as it can be seen in Figure 4.2. The service tab aims to show the user all offered services and facilities in a compact manner. Users see at first glace that they have different options to find an answer to their questions e.g. the FAQ page. Using the customer tap, topics like billing, contract alterations and benefits can be accessed. After clicking on a link provided in one of the taps e.g. the link "Samsung" in the product tap, the entire page reloads and a new page showing different Samsung devices is displayed.

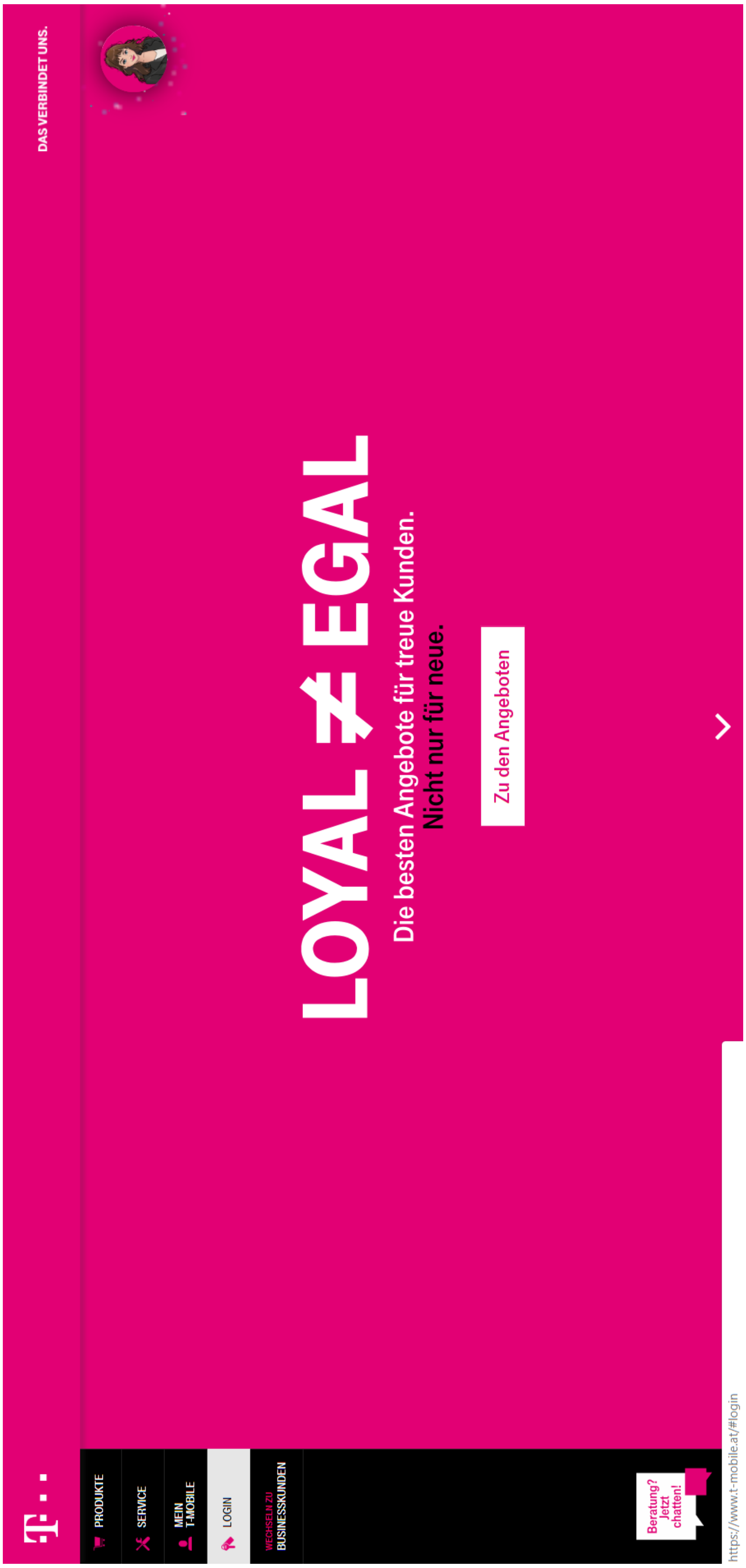

Figure 4.1: Homepage of an Austrian telecommunication company

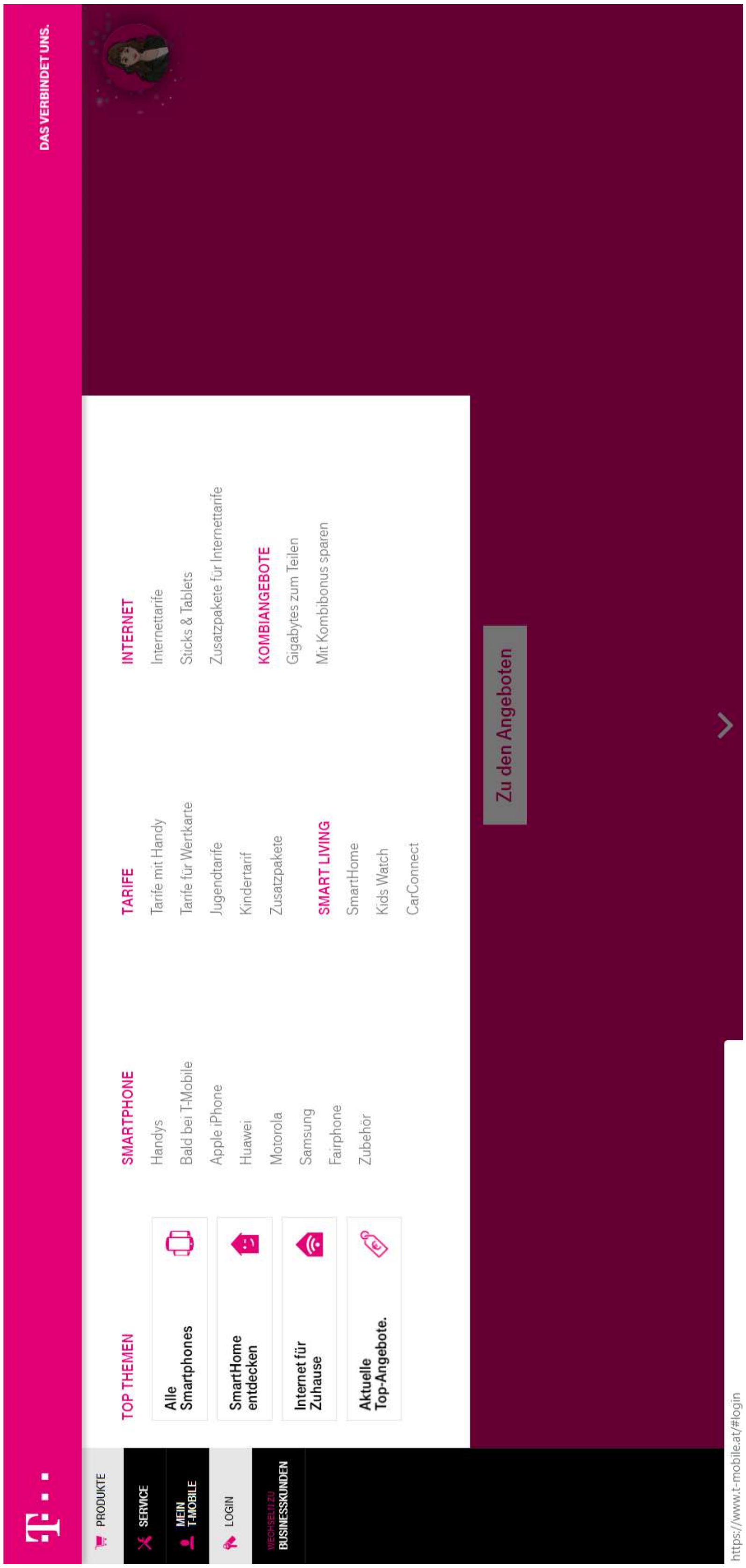

Figure 4.2: Product page of an Austrian telecommunication company

Each single observation of the dataset represents an interaction of an user with the website, e.g. a click of user u_1 on the FAQ site *https://www.t-mobile.at/faq/*. For further analysis, these five files are grouped together to one single dataset. The most relevant attributes for our analyses are the following ones:

1. Session ID
2. Request URL
3. Timestamp

Session ID

The session ID is the ID of the webtracking cookie. The cookie tracks users' entire interactions with the webpage during a session. It is important to note that this cookie cannot be used to merge users' interactions, which took place in different sessions. Therefore a user identification would be necessary, which is not available in the present dataset. Having an available attribute for grouping the interactions according to the session they belong to, statements about entire sessions as a whole can be made. In Table 4.1 a summary of websites clicked per session is presented. As there are many sessions, which contain only one click, the minimum value, first quantile, median and third quantile are equal to one. The session with the highest number of clicks has a length of 58 clicks.

Min	1st Qu.	Median	Mean	3rd Qu.	Max
1.000	1.000	1.000	1.095	1.000	58.000

Table 4.1: Length of clickstream sessions

Timestamp

The timestamp attribute captures the time when a certain website click occurs. After performing the preprocessing steps of section 4.2, it could be determined that a session lasts in average for 62.6 minutes. The longest session in the entire timeframe lasts for 398 minutes, which corresponds roughly to 6.5 hours. Table 4.2 summarizes web sessions' durations of the entire dataset.

Min	1st Qu.	Median	Mean	3rd Qu.	Max
0.1	2.9	13.5	62.6	94.4	398.3

Table 4.2: Duration of clickstream sessions (in min)

Figure 4.3 visualizes the strong centralization on small session durations. The median value for session durations (13.5 minutes) is close to the first quantile (2.9 minutes) and results in a right skewed boxplot. From this observation follows that most of the users spend only a few minutes on the webpage during a session.

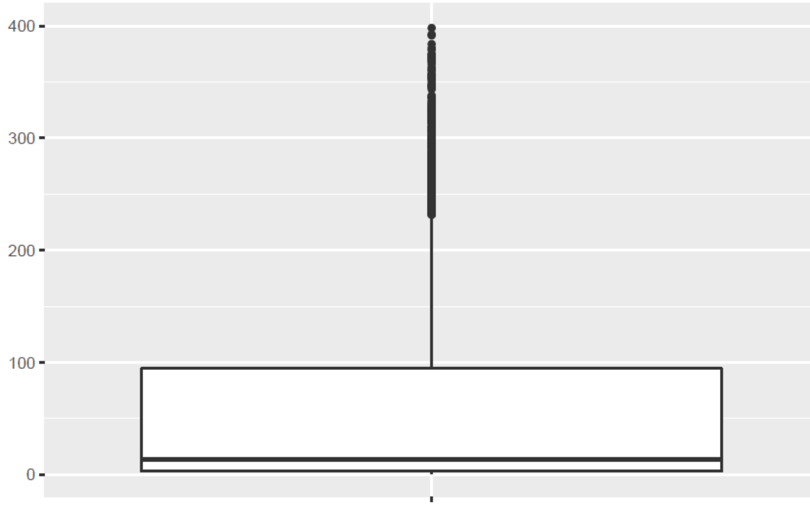

Figure 4.3: Boxplot of session durations (in min)

Request URL

The attribute request URL saves the URLs which are clicked by the users during interactions. Table 4.3 displays the top five websites, which are clicked the most as the first, second and third website during a session. The numbers are extracted after performing the preprocessing steps of section 4.2. The webpage *http://www.t-mobile.at/* is the page, which is clicked the most as the first website of a session. This site is the starting page of the company's homepage as it is displayed in Figure 4.1. Another site, which can be found under the top pages is the invoice page *https://mein.t-mobile.at/myTNT/invoice.page* where users can find their recent bills, purchases and statistics of their monthly phone calls, messages and internet consumptions. The account page (*https://mein.t-mobile.at/myTNT/account.page*) displays information regarding the user's contract agreement with the company as well as further data e.g. the 'PIN' and 'PUK' of the user. The site *https://mein.t-mobile.at/myTNT/freieinheiten* is not accessible anymore but as the link's name indicates, users can check their free minutes for calling, number of messages and data volume for the current month.

	Top 5 Websites - first Click	Number of Clicks
1	Startpage - http://www.t-mobile.at/	216
2	Startpage - https://www.t-mobile.at/	195
3	Startpage for logged in users - https://mein.t-mobile.at/myTNT/start.page	156
4	Page with most popular services - http://www.t-mobile.at/favoriten/	42
5	Invoice page - https://mein.t-mobile.at/myTNT/invoice.page	35
	Top 5 Websites - second Click	Number of Clicks
1	Startpage - https://mein.t-mobile.at/myTNT/start.page	211
2	Invoice page - https://mein.t-mobile.at/myTNT/invoice.page	100
3	Account details page - https://mein.t-mobile.at/myTNT/account.page	92
4	Startpage - https://www.t-mobile.at/	82
5	Startpage - http://www.t-mobile.at/	79
	Top 5 Websites - third Click	Number of Clicks
1	Startpage - https://mein.t-mobile.at/myTNT/start.page	155
2	Invoice page - https://mein.t-mobile.at/myTNT/invoice.page	144
3	Product page - https://mein.t-mobile.at/myTNT/product.page	110
4	Account details page - https://mein.t-mobile.at/myTNT/account.page	88
5	Overview of free units page - https://mein.t-mobile.at/myTNT/freieinheiten	66

Table 4.3: Frequently clicked websites

4.2 Data Preprocessing

Due to its unstructured form, it is necessary to preprocess the data to get it into a readable and understandable format. The following steps of data preprocessing are applied on the dataset:

1. **Separation**

   The preprocessing of data includes as a first step the separation of the dataset into separate columns e.g. session ID, timestamp and URLs, as well as a selection of attributes, which are relevant for the new dataset to be created. For the preprocessing step we decide to concentrate on the following columns for our further analysis: *session ID*, *timestamp*, and *URLs*. After splitting the data in seperate columns, we convert the timestamp, which is an UNIX timestamp, in a more readable format. E.g.: the timestamp value "1490627381" is transformed into "2017-03-27 17:09:41 CEST".

2. **Error Handling**

   Interactions having a null-value entry instead of an assigned session ID, are omitted. Otherwise it is not possible to assign these single interactions to a session and therefore to a sequence of events. In sum 8,008 out of 96,120 clicks are deleted and the remaining 88,112 clicks are used for further analysis. Repeating clicks are reduced to single clicks, e.g. if the website “http://tmobile.at” is clicked five times, this sequence of five clicks is reduced to one single click. The focus is put on sequences and the order of websites visited rather than how users behave within one site or how often they click on the same site.

3. **Matrix Generation**

   In order to better analyze the sequence of clicked websites within a session, the data is transformed to a matrix of clickstream sessions and ordered according to the websites, which the users clicked within one session. We will refer to this matrix as matrix X_{entire}. 88,112 clicks are transformed to a matrix of clickstream sessions based on their assigned session ID. The entire matrix has **74,550 clickstream sessions**. Each row of matrix X_{entire} represents one session and contains the path of URLs the users clicked within that session. The generated new matrix X_{entire} is demonstrated in table 4.4. An exemplary matrix of three sessions can be seen in table 4.4. The first column of X_{entire} contains the session ID and the other columns display the websites the user clicks sequentially within a session. In the first session ‘Session A’ the user visits the starting page of the company’s website (*https://www.t-mobile.at/*), then a page explaining the Internet options of the company (*www.t-mobile.at/myhomenet*) and finally the shop (*shop.t-mobile.at/cart/list*). In ‘Session B’ the user visits the faq page (*www.t-mobile.at/faq*) and the contract page (*www.t-mobile.at/vertrag*). Finally, in the last session of the matrix the start page, the faq page and the billing page (*www.t-mobile.at/rechnung*) are visited.

Id	1. Click	2. Click	3. Click
A	www.t-mobile.at	www.t-mobile.at/myhomenet	shop.t-mobile.at/cart/list
B	www.t-mobile.at/faq	www.t-mobile.at/vertrag	
C	www.t-mobile.at	www.t-mobile.at/faq	www.t-mobile.at/rechnung

Table 4.4: Exemplary matrix of click sequences

4. **Short Session Elimination**

   Next, sessions are deleted which contain less than three visited websites. Analyzing our dataset after the first three preprocessing steps, leads us to the decision of deleting all sessions, which have less than three clicks. The dataset containing clickstreams with as least three clicks is referred to as $X_{minThree}$.

   Table 4.5 compares the entire dataset X_{entire} to the dataset $X_{minThree}$. In both datasets the longest session counts 58 clicks, whereas the shortest session in X_{entire} is one click long and in $X_{minThree}$ three clicks. In dataset X_{entire} the minimum, first quantile, median and third quantile values are equal to one and all sessions longer than one click are interpreted as outliers due to the huge number of sessions with length one. The minimum value and first quantile of matrix $X_{minThree}$ are equal to three clicks, the median equals four clicks and the third quantile is equal to six.

We decide to base our further analyses on the dataset $X_{minThree}$. As is it defined in research question RQ2, we aim to answer the question "How can structured, non-textual website clickstreams be used to predict users' future actions?" Sessions containing one click do not give any information on the future clicks of a user. Another reason for choosing $X_{minThree}$, is the distortion of data. Due to the huge amount of clickstream sessions of length one, all sessions with more than one click are interpreted as outliers.

Dataset	Min	1st Qu.	Median	Mean	3rd Qu.	Max
$X_{minThree}$	3	3	4	4.887	6	58
X_{entire}	1	1	1	1.095	1	58

Table 4.5: Comparison of click sequence lengths: X_{entire} & $X_{minThree}$

22,471 out of 74,550 sessions from dataset X_{entire} are eliminated. The $X_{minThree}$ dataset contains in sum 52,079 sessions.

4.3 Markov Chains

Meyn and al. [MT12] define Markov chains as a collection of random variables $\Phi = \{\Phi_n : n \in T\}$, whereas T is a countable set.

Given the process Φ, which evolves on a space X and has a probability law P, there must exist a set of "transition probabilities" $\{P^n(x, A), x \in \mathrm{X},\ A \subset \mathrm{X}\ \}$ such that x, which is an element of space X, will be in set A, which is a subset of X, after n steps.

Markov Chains can be used as models for data, which consist of sequences of events. The transitions between the different states are labelled by probabilities of getting from a certain state A to another one B. Markov Chains consist of sequences of random variables.

Meyn et al. [MT12] illustrate the use of Markov chains by the example of exchange rate calculation. The exchange rate X_n between two different currencies is defined by its past k values $X_{n-1}, ..., X_{n-k}$. The current exchange rate X_n depends on the past $X_{n-1}, ..., X_{n-k}$ rates. This example results in the following model:

$$X_n = \sum_{j=1}^{k} \alpha X_{n-j}$$

Definition of Discrete-Time Markov Chains

Discrete-time Markov Chains (DTMC) have a set of states, transitions between those states and a starting state. The set of these states is called "the state space of the chain". A state s_j is reachable from a state s_i if there exists a positive probability of getting to state s_j if we start in state s_i [SKY+16].

Transition Probability

If a discrete-time Markov Chain moves from one state to another, this change is called "transition". The probability p_{ij} to move from state x_i to x_j is called the transition probability. The transition probability of a DTMC is determined as [SKY+16]:

$$p_{ij} = Pr(X_1 = x_j | X_0 = x_i)$$

This probability of moving from one state to another can be represented within a transition matrix. Matrix X is an exemplary transition probability matrix:

$$X = \begin{bmatrix} p_{1,1} & p_{1,2} & \dots \\ p_{2,1} & p_{2,2} & \dots \\ \dots & \dots & \dots \end{bmatrix}$$

The aim of using transition matrices is illustrated by the following example. If a person starts a clickstream session with the site *www.t-mobile.at/startpage*, there exists a high probability (0.65) that the next site the person will visit, is *www.t-mobile.at/faq*. The relation between these two sites and their probabilities of following each other can be displayed by the following matrix (see Table 4.6):

Site	www.t-mobile.at/faq	www.t-mobile.at/startpage
www.t-mobile.at/startpage	0.65	0.28
www.t-mobile.at/faq	0.15	0.67

Table 4.6: Transition matrix

The value at the position $p_{Startpage,FAQ}$ displays the probability of a person to visit the FAQ page next, when the person is currently at the startpage. The probability of that shift is 0.28.

Order of the Markov chain

An important characteristic of a DTMC is its order. The order m of a Markov chain determines the number of states, which are taken into consideration for defining the next state *m+1* of the sequence. In the previously mentioned exchange rate example by Meyn et al. [MT12] the order of the proposed model is equal to k, as k values are considered for the calculation of X_n.

The probability that the Markov chain is in state z after m steps is calculated by the following method: in order to reach a certain state z from the current state x after m steps, a certain intermediate state y has to be reached. As this intermediate state can be any state of the Markov chain and each of these possibilities have to be taken into account, the transition probability matrix for order m is defined as a P^m matrix [Som14].

Higher-order Transition Matrix

The transition matrix of a Markov chain of order two distinguishes from a transition matrix of an order-one Markov chain. Taking the T-Mobile websites as an example, the probabilities for a second order transition matrix are calculated as follows:

We start with a person at state *www.t-mobile.at/startpage* and are searching for the probability that the state after next this person will reach, is *www.t-mobile.at/iphone*. To answer this question we first have to consider the intermediate site after the startpage. There exist many possible states the person could reach. We will concentrate on three of them: *www.t-mobile.at/faq*, *www.t-mobile.at/service* and *www.shop.t-mobile.at*. As our Markov chain has an order of two, the next state has to be taken into account for predicting the state after next. There exist three different paths for getting to *www.t-mobile.at/iphone*:

1. *www.t-mobile.at/startpage => www.t-mobile.at/faq => www.t-mobile.at/iphone*
2. *www.t-mobile.at/startpage => www.t-mobile.at/service => www.t-mobile.at/iphone*
3. *www.t-mobile.at/startpage => www.shop.t-mobile.at => www.t-mobile.at/iphone*

In order to calculate the probability of getting to *www.t-mobile.at/iphone* in the second step, we have to calculate the following probabilities:

$$P = P_{Startpage,FAQ}P_{FAQ,IPhone} + P_{Startpage,Service}P_{Service,IPhone} + P_{Startpage,Shop}P_{Shop,IPhone}$$

Absorbing States

Absorbing states of a Markov chain are states, which never have a successor in a clickstream session. It is not possible to leave that state. The only outgoing transition link leads to the state itself and therefore the probability to stay in the same state x_i forever is $p_{ii} = 1$. A matrix is called absorbing if it is possible to get from each state to an absorbing state, no matter in how many steps [SKY$^+$16].

Predicting

One common goal of clickstream analysis is to predict the next click of the user. This click depends on the last k clicks in a k-order Markov chain. The following equation is used to predict the next click:

$$X^{(n)} = B \sum_{i=1}^{k} \lambda_i Q_i X^{(n-i)}$$

$X^{(n)}$ describes the distribution of state X at time n. Therefore we need the matrix of absorbing probabilities B and the transition matrix Q_i for lag i, where λ_i is the lag parameter. The previous k states are added up in order to predict the next click $X^{(n)}$ [SKY$^+$16].

Goodness of Fit Measures

Goodness of Fit measures are used to evaluate models by comparing their predicted values to the observed values in the available dataset. In scope of this work, three measures are used for this purpose: the Akaike information criterion (AIC), the Bayesian Information Criterion (BIC) and the log-likelihood value. Based on these values, Markov chain models of different orders are compared to each other. The goal is to choose the model with the smallest deviations between the predicted and actual values.

While comparing different models to each other regarding the AIC, the model with the smallest AIC should be preferred according to the criterion [Aka74]. Analogous to the AIC, in case of the BIC, the models with the smallest values are preferred as well. The log-likelihood measure is applied to determine optimal values for parameter estimation. The higher the log-likelihood value is, the better is the present estimation. Using the loglikelihood value as the only measure for model selection is not recommended due to the fact that it could parameterize a model too much while finding the best fitting model for the data [PB04].

4.4 Model Generation & Goodness of Fit Measures

As a first step of model generation, a subset of the entire dataset is chosen for modelling as otherwise the computing power would exceed the available capacities e.g. for calculating the transition matrices. The subset contains 300 clickstream sessions and is used for further analysis. Next, the dataset is divided into a test and a training set. Test and training observations are chosen randomly. The horizontal split of dataset $X_{minThree}$ results in two datasets: X_{test} and $X_{training}$. 80% of the original dataset are used as a training set, which means that this part of the data is used for training the Markov chain model. The resulting training dataset $X_{training}$ has in sum 240 sessions. The remaining 20%, on the other hand, are used for testing in section 4.5. The test set contains the remaining 60 sessions.

Summaries of the generated order zero, order one and order two Markov chains can be seen below in Table 4.7.

Markov Chain	States	Observations	Loglikelihood	AIC	BIC
Order zero	65	240	-942.44	1184.877	1184.877
Order one	65	240	-275.02	682.037	924.93
Order two	65	240	-275.02	814.04	1299.82

Table 4.7: Comparison of three different Markov chain models

As it is described before, the order of a Markov chain model determines how many previous states are taken into account for predicting the next state. Based on the present dataset, the models above predict the next click of a user while considering their past click history. The generated Markov chain model of order zero, which is referred to as

m_0, is the baseline model for the predictions in section 4.5. As the model has an order of zero, no past clicks are considered for prediction. In other words, the zero order model chooses the next click randomly. The first order model is named m_1 and can be applied for predicting the next state. As per definition, the second order Markov chain m_2 can predict the next click and the click after next.

The number of observations displays how many clickstream sessions have been taken into account while generating the model. As each of the three models is trained using the same dataset, each model summary counts 240 observations. The number of states counts the number of unique URLs in the dataset used for modelling. 240 observations include 65 unique states. These states are also known as "the state space of the chain" as it has been explained previously. Due to the fact that for each of the three Markov chain models the same dataset has been used for modelling, the number of states and observations are identical. Comparing the measures of goodness to each other e.g. the AIC, the BIC and the log likelihood, the Markov chain model of order one captures the dataset and the setting better than the other two models. The log likelihood values for order one and two are same, but the AIC and the BIC are smaller for the model of order one therefore this model fits better.

Transitions & Transition Matrices

As the dataset used for modelling consists of 65 states, the transition matrix of the order one Markov chain model has in total 65 rows and 65 columns. In contrast, the second order matrix is composed of 130 rows and 65 columns.

Table 4.8 demonstrates an extract of the transition matrix of model m_1. Out of the 65 x 65 matrix, a 2 x 2 subset has been extracted. The table displays that the transition probability of getting to state *mein.t-mobile.at/myTNT/account.page* when the current state is *mein.t-mobile.at/myTNT/product.page*, is equal to 47%. The transitions probabilities from *mein.t-mobile.at/myTNT/start.page* to *mein.t-mobile.at/myTNT/account.page* and from *mein.t-mobile.at/myTNT/product.page* to *mein.t-mobile.at/myTNT/invoice.page* are both equal to 16%.

	[...]/account.page	[...]/invoice.page
mein.t-mobile.at/myTNT/product.page	0.47	0.16
mein.t-mobile.at/myTNT/start.page	0.16	0.2

Table 4.8: Subset of the transition matrix of model m_1

Analyzing the transition matrices of the different Markov chain models, it becomes apparent that each state has a high transition probability to one or two other states. The transition probabilities to all other states are relatively low. In other words, many clickstream sessions have a similar sequence and a page p_0 is followed by another page p_1 whenever p_0 occurs within a session. Therefore only a few transactions within the transition matrix have a transition probability near one and the majority of transactions have a probability of next to zero. A transition with a probability value equal to one, illustrates that all sessions reaching state s_n, have s_{n+1} as their successor. After closer inspection of the pages in our transition matrix, it becomes apparent that most of the links don't exist anymore and forward to the starting page *https://mein.t-mobile.at/myTNT/start.page*.

Using the transition matrix of an order one Markov chain and the current state of a user, the next step can be easily predicted. Moreover, using the transition matrices, the states with the highest start and end probabilities can be extracted. Table 4.9 displays the start probabilities of all models. These states are the starting points of different clickstream sessions. The probabilities of the starting pages correspond to the probability that a clickstream session will start at this specific page. As all possible starting websites are displayed, the sum of their probabilities is equal to one. The website *https://mein.t-mobile.at/myTNT/start.page* has the highest starting probability of 16.7%. The product page *https://mein.t-mobile.at/myTNT/product.page*, which summarizes an user's purchased products and service packages, has the second highest start probability. Additional starting pages are the invoice page, the account page and pages giving information on Internet packages, mobile devices, payments, configurations, etc.

State	Start-Probability
Homepage - https://mein.t-mobile.at/myTNT/start.page	0.1677
Products - https://mein.t-mobile.at/myTNT/product.page	0.1333
Invoice - https://mein.t-mobile.at/myTNT/invoice.page	0.10
Free units - https://mein.t-mobile.at/myTNT/[...]=freieinheiten	0.0833

Table 4.9: Model states with highest start probabilities

Absorbing States

The previous summaries of model m_0, m_1 and m_2 tell us that the Markov chain models have absorbing states. Absorbing states are state, which do not have any succeeding state in the entire dataset. These states are called *absorbing*, because as soon as such a state is reached, the flow process ends. The absorbing states of our Markov chains are listed below:

- Network availability - http://www.t-mobile.at/netz/index.php
- Internet roaming - http://www.t-mobile.at/surfen-ohne-sorgen/pakete/index.php
- Account & contract - https://mein.t-mobile.at/myTNT/account.page
- Personal products - https://mein.t-mobile.at/myTNT/product.page
- Customer page - https://mein.t-mobile.at/myTNT/shortcut=customerNotifications
- https://mein.t-mobile.at/myTNT/portlet.page?shortcut=handy_config

- http://www.t-mobile.at/porsche-design-huawei-mate-9/index.php
- http://www.t-mobile.at/samsung-galaxy-s7-edge/

Three out of the eight links above, namely *http://www.t-mobile.at/porsche-design-huawei-mate-9/index.php, http://www.t-mobile.at/samsung-galaxy-s7-edge/* and *https://mein.t-mobile.at/myTNT/portlet.page?shortcut=handy_config*, did not exist anymore at the time of analysis as the telecommunication company was continuously updating its website. Therefore it was not possible to reproduce the content of these pages. The remaining five pages contain information about the network availability (*http://www.t-mobile.at/netz/index.php*), roaming (*http://www.t-mobile.at/surfen-ohne-sorgen/pakete/index.php*), contract details of the logged in user (*https://mein.t-mobile.at/myTNT/account.page*) and purchased products and service packages of the logged in user (*https://mein.t-mobile.at/myTNT/product.page*). The contents of these websites (network, roaming, products, billing, contract) can be understood as topics and answers users are looking for while browsing and clicking.

All absorbing states of a Markov chain are also end states but not vice versa. End states are states which occur in some sessions as last states of the sequence, but in other clickstream sessions they can have outgoing links. Absorbing states can be interpreted as pages, which give answers to users' questions and therefore the clickstream sessions end at this point of interaction. The absorbing states of a Markov chain model can provide indications of users' purposes.

Receiving a user interaction as input, which preceded an absorbing state, the system can provide the absorbing state right away. A short example demonstrates the benefits of knowing the absorbing states of a system: Clickstream c_1 starts with a user searching on the site *t-mobile.at/service*. On this site the system provides information on all services which are offered. After numerous intermediate stages, 90% of all users reach the absorbing state *https://mein.t-mobile.at/myTNT/portlet.page?shortcut=ebill* and stop searching afterwards. Out of this simple example, it can be learned that a link on the site *t-mobile.at/service* should be installed, which leads directly to the billing site. This saves time for the majority of users and increases their satisfaction.

4.5 Prediction

The succeeding examples of click prediction are calculated based on the described settings of test- and training-sets at the beginning of this chapter. Different input sequences and the following trained models are used: the Markov chain model of order zero (m_0), order one (m_1) and order two (m_2). The Markov chain models are compared to each other by comparing their predicted clicks to real world data taken from the test dataset.

Single clicks used as an input for the models and the predicted next states are compared to each other. This is done in prediction P_1. Next, two clicks are used as input and the click after these two states is predicted in P_2. The third prediction example P_3 uses *http://www.t-mobile.at/* as the current click to predict the next two clicks of the sequence. Finally, the entire test dataset is used for prediction. In prediction example P_4 the first click of each clickstream session is used as input for the prediction models. The next click for each clickstream of the test dataset is predicted and compared to the observed value. Out of this comparison a percentage of correctly predicted clicks is calculated.

Prediction 1

Past Click: Start page - *http://www.t-mobile.at/*

- Prediction result for model m_0: *https://mein.t-mobile.at/myTNT/start.page* (15.6%)
- Prediction result for model m_1: *https://mein.t-mobile.at/myTNT/start.page* (27.3%)
- Prediction result for model m_2: *https://mein.t-mobile.at/myTNT/start.page* (27.3%)

Given the sequence s_1 with the starting state *http://www.t-mobile.at/* as an input, the Markov chain models of order-zero, order-one and order-two are utilized to predict the next step of this clickstream sequence. In each of the three cases we receive the website *https://mein.t-mobile.at/myTNT/start.page* as a result. Furthermore the prediction results that this page is not an absorbing page.

As described previously, a zero order Markov chain is determined by the fact that it is independent of the present and previous steps of the Markov chain and each next step is chosen randomly. Given the sequence *http://www.t-mobile.at/*, we predict the next step of our clickstream session and receive the website *https://mein.t-mobile.at/myTNT/start.page* as a result. The probability of reaching this next step is 15.6%.

The order one Markov chain m_1 predicts the same following click *https://mein.t-mobile.at/myTNT/start.page* while starting with sequence s_1. Using an order one Markov chain model we receive the probability of 27.3% as this model has considered the last click of the sequence: *http://www.t-mobile.at/*.

The order-two Markov chain m_2 gives us the same result as the order one model. Markov chains of order two utilize not only the present state of the input sequence but also the next click to predict the click after next. As our input sequence $s_1$has only one click, this is the only state which can be considered by the order-two Markov chain model. Therefore the result does not distinguish itself from the previous result using m_1. The probability of this predicted sequence is also 27.3%.

Prediction 2

Past Clicks: *{http://www.t-mobile.at/, https://mein.t-mobile.at/myTNT/start.page}*

- Prediction result for model m_0: *https://mein.t-mobile.at/myTNT/start.page* (15.7)
- Prediction result for model m_1: *https://mein.t-mobile.at/myTNT/invoice.page* (20.4%)
- Prediction result for model m_2: *https://mein.t-mobile.at/myTNT/invoice.page* (25%)

Predicting with the order-zero model m_0 and the past sequence s_2: *{http://www.t-mobile.at/, https://mein.t-mobile.at/myTNT/start.page}*, we receive *https://mein.t-mobile.at/myTNT/start.page* as a result and a prediction probability of 15.7%.

Using the sequence s_2 as input for the models, the order-one model predicts page *https://mein.t-mobile.at/myTNT/invoice.page* to be the next step of the sequence. This page is predicted with a probability of 20.4%.

The second order Markov chain models, predicts the same next click as the first order model. Though, the probability of this prediction (25%) is higher in case of model m_2.

Prediction 3

Past Click: *http://www.t-mobile.at/*

- Prediction result for model m_0: *https://mein.t-mobile.at/myTNT/start.page, https://mein.t-mobile.at/myTNT/start.page* (2.5%)
- Prediction result for model m_1: *https://mein.t-mobile.at/myTNT/start.page, https://mein.t-mobile.at/myTNT/invoice.page* (6.8%)
- Prediction result for model m_2: *https://mein.t-mobile.at/myTNT/start.page, https://mein.t-mobile.at/myTNT/invoice.page* (6.8%)

Next, we use the same past click sequence as in the first prediction (s_1) as model input. The aim is to predict the next two clicks of a user. Model m_0 predicts the following predicted sequence: *https://mein.t-mobile.at/myTNT/start.page, https://mein.t-mobile.at/myTNT/start.page*. The probability we receive for this prediction is equal to 2.5%.

Starting at the page *http://www.t-mobile.at/* again and looking for the next two steps of our chain, we receive the following predicted sequence: *https://mein.t-mobile.at/myTNT /start.page, https://mein.t-mobile.at/myTNT/invoice.page* while using model m_1. This sequence is predicted with a percentage of 6.8%. In this case the predicted sequence divers from the prediction result of a zero order Markov Chain.

Using the same previous click sequence *http://www.t-mobile.at/* and the order two Markov chain, the model predicts the following sequence: *https://mein.t-mobile.at/myTNT/start .page, https://mein.t-mobile.at/myTNT/invoice.page* with a probability of 6.8%.

Prediction 4

Past Clicks: 1. click of each clickstream session

- Prediction result for model m_0: 11.7% correctly predicted
- Prediction result for model m_1: 51.7% correctly predicted
- Prediction result for model m_2: 48.3% correctly predicted

Next, we predict based on the entire test dataset. Taking the first click of each dataset session, we predict the second click for each session. The predicted second click is compared to the actual second page of the dataset. As a result we calculate the share of correctly predicted clicks.

Using a Markov chain of order zero, for 11.7% sessions the next click is predicted correctly. Model m_1 predicts 51.7% of the preceding clicks correctly and model m_2 43.3%.

Prediction 5

Past Clicks: 2. click of each clickstream session

- Prediction result for model m_0: 0% correctly predicted
- Prediction result for model m_1: 43.3% correctly predicted
- Prediction result for model m_2: 38.3% correctly predicted

The same prediction is conducted as in the example before. This time the second click of each session is taken as model input to predict the third click.

Markov chain model m_0 is not able to predict any of the next clicks correctly and delivers a result of 0%. Again model m_1 results in a higher percentage of correct predictions (48.3%) than model m_2 (38.3%).

The predictions result that the order one Markov chain m_1 delivers the highest prediction rates for predicting the next click of the user. Using the second order Markov chain and a single click for prediction, m_2 predicts with similar or lower prediction rates compared to model m_1. Taking two clicks as input for prediction, the second order Markov chain predicts with the highest probability rate. This is attributable to the fact that model m_2 is the only model capable of taking two past clicks as input whereas model m_1 can only predict based on the current state and model m_0 predicts randomly without considering input information e.g. the current state of the sequence.

4.6 Summary

As a first step of model development, the available dataset is preprocessed according to the steps described in 4.2. As the result of the preprocessing steps, a dataset of clickstream sessions is generated. Table 4.4 displays a subset of this dataset where each entry equals one clickstream session. The first column saves the session ID and the remaining columns the clicks, which belong to this session. Due to the fact that a huge amount of sessions contain only on click and increase the biasness of the analyses, these sessions are excluded. The analyses of the chapter are based on the dataset $X_{minThree}$, where each clickstream session consists of at least three clicks.

Based on the preprocessed dataset, Markov chain models of order zero, one and two are built. The entire dataset is divided in a test set, containing 20% of all sessions, and a training set (80%). The training dataset is used for training the models. The prediction results of the first and second order models are compared to each other and to the zero order Markov chain model, which is used as a baseline. The clicks used as inputs for the models, are taken from the test dataset.

The model evaluation consist of three main stages:

1. Statistical evaluation of the model
2. Comparison of predicted behavior to real world data
3. Evaluation with the aid of domain experts

First, the models are evaluated statistically by comparing measures of goodness to each other. The Akaike Information Criterion (AIC), the Baysian Information Criterion (BIC) and the loglikelihood are taken into account. The baseline model has the highest AIC and BIC values on the one side and the smallest value for the loglikelihood measure on the other side. Each one of the three measures illustrates that this model fits the real world data the least and the deviations between the predicted and actual clicks are the highest. Comparing the first and second order model to each other, both models have the same loglikelihood value (-275.0185). Therefore, the AIC and BIC measures are compared to each other. The first order model has smaller values for the AIC (682.037) and the BIC (924.9284) than the second order model (AIC: 814.037; BIC: 1299.82). Based on the statistical evaluation and the measures of goodness, the Markov chain model of order one is preferred.

Secondly, the models are compared and evaluated based on their prediction accuracy. Five different prediction examples are presented. The first three examples are based on single sites, which are used as inputs for the models to predict the next site or the next two sites. The last and second last prediction examples take the entire test dataset as inputs.

In prediction P_4 the entire first column of the test dataset is used as model input and the second click of each session is predicted. Next, the predicted values are compared to the observed values and the share of correctly predicted sessions is calculated. The first order Markov chain model predicts 51.7% of all clicks correctly and the second order model 48.3%. The baseline model m_0 predicts only 11.7% of all clicks correctly. The last prediction example P_5 uses the second click of each session as input and predicts the third click based on the entire test data. Model m_0, which chooses random values for prediction, has 0% prediction accuracy. Whereas the first order model predicts 43.3.% correctly and the second order Markov chain 38.3%. Both, the first and second order model perform always better than the baseline model. Comparing the prediction results of the Markov chains to each other, it can be summarized that the order one Markov chain m_1 delivers the highest prediction rates for predicting the next click. Using the second order Markov chain and a single click for prediction, m_2 predicts with similar or smaller prediction rates compared to m_1. Taking two clicks as input for prediction, the second order Markov chain predicts with the highest probability rates. This is attributable to the fact that model m_2 is the only model capable of making predictions not only based on the current state of the sequence but also considering the past state.

In the third place, the Markov chain models are evaluated with the help of the cooperation partner. Experts of the telecommunication domain accepted the technique as a generic method, which is easily reusable and applicable for them as well.

CHAPTER 5

Discussion

The analysis of the previous chapters has been done to address the challenge of understanding users of the telecommunication domain with their information needs, preferences and their satisfaction. The aim of this thesis is specified by the following research questions:

RQ1 "How can (semi)structured, textual chat data of users be used to model and understand users' information needs, preferences and satisfaction in the telecommunication area?"

RQ2 "How can structured, non-textual website clickstreams be used to predict users' future actions?"

In this chapter we first discuss the insights gained from exploratory analysis, text mining and event sequence analysis using the chatbot conversation data to answer the first research question. Secondly, the outcomes of exploratory clickstream analysis and Markov chain modelling are discussed in a similar manner for the second research question. Finally, the usage of Markov chains to predict users' future click behavior is analyzed.

5.1 Chatbot Analysis

Analyzing users' interaction times with the chatbot shows that the highest number of interactions between users and the chatbot does not take place in the evenings as one might suppose, but in the traditional office hours between 9 am and 5 pm (see Figure 3.2). This can be related to the fact that during office time it is easier for a person to chat than to make a phone call and talk to a telecom service employee. Users who take the chatbot as an alternative for talking with a human salesperson might face a number of restrictions e.g. they don't have enough time for making a call or cannot talk for whatever reason. Figure 3.3 is in line with these findings as it shows that more than 80% of conversations took place in the weekdays and only 19% of chats were written on

Saturdays or Sundays. The majority of users like to chat with the telecommunication chatbot during the office hours and on weekdays. As time capacities during the week in daytime are usually more restricted, it is becomes even more important that the chatbot can provide accurate answers to user needs quickly. The durations of conversations support this argument as it can be seen in Table 3.3. More than half of the conversations end within one minute. A strong focus has to be put on determining users' needs quickly in conversations before they quit chatting.

Valuable information about users' needs and topics they ask the most questions about, can be extracted by analyzing the categories the conversations have been labelled with. Approximately 16% of all conversations have been labelled with a certain chat topic by T-Mobile. The majority of labelled conversations deal with questions about users' invoices (approx. 19%), as well as T-Mobile's Internet services (15%) and email services (12%). Still, these insights are limited as the 84% of conversations are not labelled at all. It is necessary to also consider these conversation for extracting users' needs as otherwise valuable information would be ignored. The topics 'invoice', 'Internet', 'email' and 'un-/locking of SIM cards' can be also found in unlabelled conversations as it can be seen in the tables 3.14, 3.16 and 3.17. These findings emphasise the insights on users' information needs based on the labelled data. Additionally to these categories two further chat topics become apparent: contacting customer service and contract modifications. The bigram, which occurs the most in users' questions is 'kundenservice kontaktieren' (contacting the customer service). In the list of frequently occurring terms in users' questions tokens such as 'Kündigung' (cancellation), 'Kündigungsfrist' (cancellation period), 'Verlängerung' (extension) and 'Vertrag' (contract) occur. These two conversation topics (customer service contact & contract modification) are not part of the top categories according to labelling but seem to be relevant topics for users according to the conversation logs.

Overall, it can be said that valuable information needs of users can be extracted based on their chat conversations with the chatbot. Labelling conversations manually helps to understand the underlying needs easily but it would be too time consuming labelling all chat conversations by humans. Therefore automated methods such as frequent terms and n-grams extractions have been used to determine users' information needs based on unlabelled data. On the one side, the extracted topics correspond to the labelled ones and on the other side further topics are found. These methods for topic extraction can be applied to various chatbot data independent of the domain the chatbots are developed for. Based on this information it is also possible to decide on which topics to put the focus while further developing the chatbot.

After interacting with the chatbot, users have the possibility of giving feedback to the chatbot using feedback scores. Comparing the two feedback scales which were available for users, it is recommendable to the telecommunication company and in general to everyone using chatbots for customer interaction to give their users only the possibility of rating based on a dichotomous scale {-1, 1}. On these scales users have to choose between 'like' and 'dislike' for rating. The user is forced to express his or her (dis-)satisfaction by selecting one of the two options. The Likert scale gives users the opportunity of avoiding to take a position as it is possible to rate an interaction with three stars. The rating of three stars neither express satisfaction, nor dissatisfaction. It was mentioned before that the majority of users give negative feedback. When someone is satisfied with the offered service, the incentives of giving feedback are much lower. Besides finding methods of gathering implicit feedback (e.g. event sequence analysis) the feedback scores were split according to the topics of their conversations. As the share of negative feedback is higher for the chat topic 'email', than for 'invoice' and 'homenetbox', this topic should be improved first. It is not possible for the provider of a chatbot to improve all answers at the same time. In this case the detailed analysis of explicit feedback as it has been done for the present chatbot, helps to determine on which topics the focus should be put first. Moreover, users' satisfaction with different products and services of the telecom provider can be determined.

Gaining further insights of users' satisfaction based on their feedback comments has been more challenging than expected. The majority of feedback comments consist of statements such as 'the answer was wrong.' or 'the answer did not help me.' and vulgar language is often used to express ones dissatisfaction. Users prefer to give feedback scores rather than writing that the answer was dissatisfying. Moreover, further information on reasons for dissatisfaction or why the given answer did not match is not provided. We also tried to extract feedback comments according to chat topics, which has proved to be difficult and not informative due to the small amount of labelled feedback comments (see Table 3.19).

Finally, it has to be pointed out that explicit feedback cannot be used as single source of information for determining user satisfaction. Users tend to give feedback more often when they are dissatisfied than in case of satisfaction therefore feedback is mostly negatively connoted. Another way of analyzing satisfaction and the fulfillment of needs, is to observe users' implicit feedback i.e. analyzing their behavior after they leave the chat. A user, who continues to search for an answer or calls a customer service employee after chatting, is probably not satisfied with the received answer. Due to the available data, it is not possible to analyze behavior after chatting.

The measure q_n has been introduced in the Chapter 3 as a value to evaluate predefined chat conversations and their answers, which are matched to users' questions. The resulting boxplot of q_n values shows (see Figure 3.10) that the majority of chatbot conversations continue after an answer is given. Based on the list of answer nodes with the highest number of outgoing edges (see Table 3.21) and smallest q_n values, wrongly installed answer nodes can be determined and remodelled. A website containing information on automated bank invoice collection by the telecom provider has the smallest q_n value of 0.5. During all of the 21 conversations, the chat did not end when this website was given as an answer. As a next step of error handling, it is suggested to analyze the 21 conversations where this answer was given and to adjust the predefined, underlying conversations. Overall, the distribution towards small q_n values suggests that the use of predefined conversations for interacting online with telecommunication users should be reconsidered and adjusted. Based on the amount of available data on user - chatbot interactions, more accurate and recent methods for question answering could be developed in comparison to rule based answering.

Overall, the discussed methods for chatbot analysis can be applied to any chatbot independent of the domain the chatbot is developed for. These insights are especially important for evaluating and improving chatbots which are already in use for interacting and communicating with users. Developing chatbots or other tools for user interaction based on abstract concepts e.g. predesigned chatbot conversations might be necessary at an early state of development but these concepts have to be evaluated as soon as real data is available. Moreover, we have shown that in our use case explicit feedback is not enough for evaluating the chatbot. Firstly, users usually don't like to give feedback and when they give feedback, the majority of users only communicates that they are not satisfied without providing further information. For this reason, the proposed methods for extracting implicit feedback out of conversations become more important for evaluating chatbots.

5.2 Clickstream Analysis

In their experimental study Mandel and Johnson [MJ02] demonstrate that the design of a webpage has an influence on the choices users' make on this page. The clickstream analysis conducted in section 4 of this work aims to first of all analyze users' behavior on the website of a telecommunication company. Moreover, we predict users' future clicks based on the their past behavior on the website using Markov chain models.

Developing an efficient website structure where it is easy for users to find what they are looking for, can increase users' satisfaction. Investigating the top T-Mobile websites, which are visited by users as the first, second or third site during a session, it is evident that topics such as billing, invoice and account information are of particular importance. Users who log in to the website with their phone numbers and credentials are particularly interested in questions regarding their recent bills, their free units for calling, messaging and Internet, as well as account settings (see Table 4.3 and 4.9). Generating a transition matrix based on clickstream data (see Table 4.9) increases the readability and understandability of click sessions and transforms them into a machine processible matrix representation. Furthermore, we can determine which websites have huge transition probabilities among them, e.g. a quarter of all users visit the invoice site after opening the telecommunication website.

Using this knowledge and information on how many sites users actually have to click through in order to reach their final goal, the telecommunication company can evaluate its website's design and adjust links between websites. At the starting page of T-Mobile's website the following website links are presented right at the top: 'My Bill', 'My Contract', 'My Data', 'My Free Units', 'Contact' and 'Frequently Asked Questions'. After logging in to the telecom website, a website is displayed for the company's customers with three huge button directing to the sites 'personal data', 'bills' and 'contract & agreements'. It is clearly visible that the telecommunication company has developed this website based on users' click behavior data and it is aligned with users' needs.

The absorbing states of our clickstream data include the company's product page as well as two mobile phone pages. It can be assumed that users who stop their interaction process at these pages were looking for new mobile products sold by the company and after reaching the desired page they stopped their (click-) search process. As the company's product page and account page are also part of the absorbing states list, the insights gained before from the transition matrices are underlined. It was mentioned that due to the huge amount of transitions leading to these pages, they can be suggested together with a few further websites (e.g. the invoice page) right at the beginning of each user session. Not only the absorbing state of Markov chains but also the end state of each click session contains useful information about users' needs e.g. many clickstream sessions ending at the invoice and contract page emphasis that a significant part of users probably visit the website because they need information about their recent bills and contract details.

Finally, the trained Markov chain models were used for predicting users' future clicks based on their past behavior. Approximately half of the click sequences of the test dataset could be predicted correctly. As the majority of users is looking for the same topics and answers (e.g. invoice, billing, Internet problems), predicting users future clicks is easy. Due to the reason that users' behavior is foreseeable, the system can predict their future steps and recommend answers to users' questions right at the beginning of the session. By doing this, search processes can be reduced and satisfaction can be improved. It was discussed before that users' satisfaction is correlated to the time needed for finding the answer to their problem. Moreover, the website of the telecommunication company can be redesigned so that it aligns to users' natural click behavior. It is relevant to mention that the first order Markov chain performs the best in respect to the available clickstream data. One might expect the second order Markov chain to perform better as m_2 not only considers the current state of the click sequence but also the last one. Our analysis shows that the selection of clickstream models has to be aligned to the structure of available data in order to reach best prediction results.

Overall, a major advantage of clickstream analysis is the fact that it is widely applicable to different domains and websites. In contrast to the chatbot data analysis conducted previously, the insights on clickstream analysis can be translated for every company which interacts with their users over its website. The only requirement is the collection of clickstream data using a cookie, which most websites have already integrated beforehand. Moreover, the format of the data necessary for clickstream analysis is very simple. It only requires the websites' URLs to be sorted in order of visit per session as it can be seen in Table 4.4.

CHAPTER 6

Conclusion

6.1 Summary

The main goal of this thesis is to address the challenge of understanding users of the telecommunication domain with their information needs and preferences. Based on this knowledge, personalized offers of services and products can be generated.

First, a literature overview on the state of the art in the areas recommender systems, chatbots and clickstream analysis was given. Recommender systems, which are well established technologies in other domains e.g. travel and retail, were introduced with reference to their usefulness in the telecommunication domain. Although the telecommunication domain has access to a huge amount of data, the available information is not used to the full extend by the majority of telecom providers. Studies conducting pioneering work in the area of user interaction and telecommunication were presented. Furthermore, the importance of user modelling as a first step of understanding users was highlighted. As it is pointed out in the research questions, the aim of this work is to better understand and model users of the telecommunication domain, including their information needs, preferences and satisfaction.

The data used for analysis in scope of this work, was provided by the Austrian telecommunication company T-Mobile AT. At a first meeting with the cooperating partner, insights and questions regarding the provided data were discussed. Based on the available data, the further work can be separated into two parts: chatbot analysis and clickstream analysis.

Chatbots, as an emerging topic in research and industry, aim to increase users' satisfaction by more information availability and time savings. As no common understanding of the term "chatbot" exists in industry and research, it is important to mention that the telecommunication chatbot analyzed in this work is a task-oriented, knowledge based system. The aim of the chatbot is to answer user questions regarding the products and services of the telecommunication company. The provided chatbot data was categorized into three groups: basic attributes, attributes used for text mining and attributes used for event sequence analysis.

After conducting descriptive analysis, the data was preprocessed for text mining and text corpora were generated. The corpora were used to determine the topics of the conversations. Next, the labelling of questions and answers, as well as URL links and FAQ links were used to generate a new dataset for event sequence analysis which was then used to model chat conversations as networks. Each node of the network represents one event e.g. asking a question about billing or clicking on a recommended link. The directed edges between the nodes illustrate the orders of visiting nodes. Using the information of ingoing and outgoing links per node, it was calculated and evaluated if the answer nodes are installed correctly. The the introduced measure q_n determines how many users continue chatting with the chatbot after reaching a particular node. Calculating the answer qualities of each node, a boxplot was generated to display that the majority of nodes is not installed correctly (3.10).

As it is discussed in depth in chapter 5, it is possible to provide statements and results about users' topics of interests, their information needs, preferences and satisfaction based on the analysis done in chapter 3.

In Chapter 2 a literature overview on the use of clickstream data for analyzing users' web browsing behavior is provided. Clickstream data is defined as sequences of click events, which track users' click behavior using a cookie. For modelling and analyzing clickstream data, we decided to use a click sequence model, which concentrates on the clicked links and ignores further attributes. This model was selected in order to generate a model which is easily applicable for other datasets and domains. Finally, the Markov chain model was selected, as it is a well established and commonly used method for click sequence modelling. The preprocessed dataset was used to generate a test dataset (20%) and a training dataset (80%). The training dataset was used to model Markov chain models of order zero, one and two.

The models were evaluated in three steps. The first evaluation was done by comparing the models' measures of goodness. The Markov chain model of order one showed the best fit in comparison to the other models according to the AIC, the BIC and the loglikelihood. Secondly, the test data was used to compare the models regarding their prediction accuracy. Considering the prediction test cases, the first order model resulted the highest accuracy while predicting the next click. The prediction accuracy of the second order Markov chain was the highest when two clicks were used as input for prediction. Finally, the applicability of the approach was evaluated with the help of experts of the telecommunication domain. The experts classified the Markov chain approach as a widely applicable and easily understandable approach. Using the Markov chain models, further knowledge about users' click behavior could be extracted based on the transition matrices. It was determined that each site has a high transition probability to few other states. In other words, visiting site 'A' during a clickstream session, primarily results in visiting of site 'B' as the next click. All other possible states, which could be visited after 'A' have a low transition probability and were either never visited or just in case of a few clickstream sessions.

6.2 Limitations & Future Work

One of the major limitations we faced during the work of this thesis, was the data availability e.g. it was not possible to evaluate the chatbot entirely as no information on user behavior after the chat was available. It can be proposed that tracking users across longer periods of time and across different points of interaction is necessary to generate an integrated user model. This model should be updated with each preceding interaction with the user. Evaluating the chatbot conversations is difficult as we do not know what the user did after leaving the conversation. A user who continues to search on the website or calls a customer service employee, is probably not satisfied with the chatbot's answer. This information is important to determine the impact of the chatbot's recommendations. Furthermore, many URLs of the clickstream data and the sites where chatbot conversations start, include "/myTNT/" in their link. These links imply that at the time of interaction, the user was logged in on the website using the personal mobile number and a password. Access to further user information i.e. contract details or age should be available but are not provided in the dataset by the telecommunication company. Data limitations were also faced during clickstream analysis as many URLs of the dataset led to non-functional sites. This happened due to the reason that the website was further developed and URL links were changed during our analysis process. Reconstructing users' behavior was often difficult as knowledge about the content of the sites was missing.

In order to develop a comprehensive user model, a complete dataset containing chat data, clickstream data, contract details and transcriptions of phone calls with the customer service is indispensable.

APPENDIX A

Appendix

Sperren1	Homenetbox
Zusatzpakete	E-Mail
Roaming1	Rechnung
Sperren 2	Roaming4
Störung	Wie geht es dir?
Roaming2	Roaming3
Automatischer Sprachservice (Vertragskunden)	de
Rücksendung bei Defekt	Rücksendung
Automatischer Sprachservice (Klax)	Handyhilfe
Rechnung2	LTE
Urheberrechtsabgabe	Mehrwertdienste
Basispaket	Freieinheiten im Kündigungsmonat
Verrechnung der monatlichen Grundgebühren	Hardwarefinanzierung
Infodienste	Teilzahlungsservice
Manuelle Bezahlung von Rechnungen	Zahlungsarten
e-Rechnung	HomeNetbox2
Einzelgesprächsnachweis	Erstellung der Endabrechnung
EPS Onlinezahlung	Kundenkennwort und Benutzerkennwort
Sim-Karte - [...] Bestellung bis zum Tausch	Telefonbucheintrag
T-Mobile Internetschutz	Abfrage der Freieinheiten
Multi-Sim	PIN und PUK
FaceTime	Simlock für iPhones bei T-Mobile
Simlock für Android-Geräte bei T-Mobile	Mahnstopp
Das automatisierte Mahnwesen	iMessage
Die Mein.T-Mobile App	Vertragsübernahme
Sachwalterschaft	Fair und Sicher
Vertragsanmeldung	Handyschutz alt
Alte T-Mobile-Tarife	Erstellung der Endabrechnung FAQ
Freizeichentöne	Bonus-SIM
Schuldenregulierungsverfahren	DEEZER und Electronic Beats
Mehrwertdienste_FAQ	Anklopffunktion
Ablauf Mahnwesen und Gebühren	Automatische und manuelle Konfiguration

Table A.1: List of total 64 categories, Tinka sessions can be assigned to

List of Figures

List of Tables

Bibliography

[Aka74] Hirotugu Akaike. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723, 1974.

[AP10] Jeremy Adler and Ingela Parmryd. Quantifying colocalization by correlation: the pearson correlation coefficient is superior to the mander's overlap coefficient. *Cytometry Part A*, 77(8):733–742, 2010.

[AS07] Elaine Allen and Christopher Seaman. Likert scales and data analyses. *Quality progress*, 40(7):64–65, 2007.

[BCHC09] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer, 2009.

[Boi18] Nathaniel Boisgard. State-of-the-art approaches for german language chatbot development. Diploma thesis, Vienna University of Technology, 2018.

[Bur02] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002.

[Bur07] Robin Burke. Hybrid web recommender systems. In *The adaptive web*, pages 377–408. Springer, 2007.

[CHN03] Patrali Chatterjee, Donna L. Hoffman, and Thomas P. Novak. Modeling the clickstream: Implications for web-based advertising efforts. *Marketing Science*, 22(4):520–541, 2003.

[DPM+13] Bidyut Das, Subhajit Pal, Suman Kr Mondal, Dipankar Dalui, and Saikat Kumar Shome. Automatic keyword extraction from any text document using n-gram rigid collocation. *International Journal of Soft Computing and Engineering (IJSCE)*, 3(2):1–5, 2013.

[EM04] Martin Eppler and Jeanne Mengis. The concept of information overload: A review of literature from organization science, accounting, marketing, mis, and related disciplines. *The information society*, 20(5):325–344, 2004.

[Gam04] Michael Gamon. Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings of the 20th international conference on Computational Linguistics*, page 841. Association for Computational Linguistics, 2004.

[GÖ03] Şule Gündüz and M Tamer Özsu. A web page prediction model based on click-stream tree representation of user behavior. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–540. ACM, 2003.

[GRM15] Wilfried Grossmann and Rinderle-Ma. Data mining for temporal data. In *Fundamentals of Business Intelligence*, pages 207–244. Springer, 2015.

[GS00] Jeremy Goecks and Jude Shavlik. Learning users' interests by unobtrusively observing their normal behavior. In *Proceedings of the 5th international conference on Intelligent user interfaces*, pages 129–132. ACM, 2000.

[HBZG14] Mohamed Ramzi Haddad, Hajer Baazaoui, Djemel Ziou, and Henda Ben Ghezala. A predictive model for recurrent consumption behavior: An application on phone calls. *Knowledge-Based Systems*, 64:32–43, 2014.

[HFMO10] Heath Hohwald, Enrique Frías-Martínez, and Nuria Oliver. User modeling for telecommunication applications: Experiences and practical implications. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 327–338. Springer, 2010.

[IAKH18] Yuichiro Ikemoto, Varit Asawavetvutt, Kazuhiro Kuwabara, and Hung-Hsuan Huang. Tuning a conversation strategy for interactive recommendations in a chatbot setting. *Journal of Information and Telecommunication*, pages 1–16, 2018.

[JZFF10] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. Introduction to basic concepts. In *Recommender systems: an introduction*, pages 13–50. Cambridge University Press, 2010.

[Ken14] Graeme Kennedy. Chapter one: Introduction. In *An introduction to corpus linguistics*, pages 1–12. Routledge, Taylor & Francis Group, 2014.

[KH18] Julian Kawohl and Stephanie Hass. Customer service 4.0 - wie gut sind chatbots. https://www.heise.de/downloads/18/2/5/4/1/3/4/2/Studie_chatbots.pdf, 2018. Accessed: 2018-11-25.

[KZM16] Yiming Kong, Hui Zang, and Xiaoli Ma. Human network usage patterns revealed by telecom data. In *Big Data (Big Data), 2016 IEEE International Conference on*, pages 1621–1626. IEEE, 2016.

[LFB+17] Minha Lee, Lily Frank, Femke Beute, Yvonne de Kort, and Wijnand IJsselsteijn. Bots mind the social-technical gap. In *Proceedings of 15th European Conference on Computer-Supported Cooperative Work-Exploratory Papers*. European Society for Socially Embedded Technologies (EUSSET), 2017.

[Lik32] Rensis Likert. A technique for the measurement of attitudes. *Archives of psychology*, 1932.

[LPSH01] Juhnyoung Lee, Mark Podlaseck, Edith Schonberg, and Robert Hoch. Visualization and analysis of clickstream data of online stores for understanding web merchandising. *Data Mining and Knowledge Discovery*, 5(1-2):59–84, 2001.

[MJ02] Naomi Mandel and Eric J Johnson. When web pages influence choice: Effects of visual primes on experts and novices. *Journal of consumer research*, 29(2):235–245, 2002.

[MJ09] James H Martin and Daniel Jurafsky. In *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall, 2009.

[MSB+14] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.

[MT12] Sean P Meyn and Richard L Tweedie. Communication and regeneration. In *Markov chains and stochastic stability*, pages 1–168. Springer Science & Business Media, 2012.

[MWHZ13] Jun-Ki Min, Jason Wiese, Jason I Hong, and John Zimmerman. Mining smartphone data to classify life-facets of social relationships. In *Proceedings of the 2013 conference on Computer supported cooperative work*, pages 285–294. ACM, 2013.

[New18] Mark Newman. Mathematics of networks. In *Networks*. Oxford university press, 2018.

[NSSW14] Julia Neidhardt, Rainer Schuster, Leonhard Seyfang, and Hannes Werthner. Eliciting the users' unknown preferences. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 309–312. ACM, 2014.

[NTW18] Iulia Nica, Oliver A Tazl, and Franz Wotawa. Chatbot-based tourist recommendations using model-based reasoning. In *20 th International Configuration Workshop*, page 25, 2018.

[ÖK10] Özcan Özyurt and Cemal Köse. Chat mining: Automatically determination of chat conversations' topic in turkish text based chat mediums. *Expert Systems with Applications*, 37(12):8705–8710, 2010.

[OTB+14] Francisco Villarroel Ordenes, Babis Theodoulidis, Jamie Burton, Thorsten Gruber, and Mohamed Zaki. Analyzing customer experience feedback using text mining: A linguistics-based approach. *Journal of Service Research*, 17(3):278–295, 2014.

[PB04] David Posada and Thomas R Buckley. Model selection and model averaging in phylogenetics: advantages of akaike information criterion and bayesian approaches over likelihood ratio tests. *Systematic biology*, 53(5):793–808, 2004.

[PBMW99] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[PCW00] Randy Kerber Thomas Khabaza Thomas Reinartz Colin Shearer Pete Chapman, Julian Clinton and Rudiger Wirth. Crisp-dm 1.0 step-by-step data mining guide. ftp://ftp.software.ibm.com/software/analytics/spss/support/Modeler/Documentation/14/UserManual/CRISP-DM.pdf, 2000. Accessed: 2018-09-20.

[RB98] Martin Rajman and Romaric Besançon. Text mining: natural language techniques and text mining applications. In *Data mining and reverse engineering*, pages 50–64. Springer, 1998.

[RC97] David Rawlings and Vera Ciancarelli. Music preference and the five-factor model of the neo personality inventory. *Psychology of Music*, 25(2):120–132, 1997.

[RRS15] Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: Introduction and challenges. In *Recommender systems handbook*, pages 1–34. Springer, 2015.

[Sch04] Barry Schwartz. The paradox of choice: Why more is less. Ecco New York, 2004.

[SKN05] Sylvain Senecal, Pawel J Kalczynski, and Jacques Nantel. Consumers' decision-making process and their online shopping behavior: a clickstream analysis. *Journal of Business Research*, 58(11):1599–1608, 2005.

[SKY+16] Giorgio Spedicato, Tae Seung Kang, Sai Bhargav Yalamanchi, Deepak Yadav, and Ignacio Cordon. The markovchain package: A package for easily handling discrete markov chains in r. https://cran.r-project.org/web/packages/markovchain/vignettes/an_introduction_to_markovchain_package.pdf, 2016. Accessed: 2018-07-14.

[Som14] Markus Sommereder. Basic modelling techniques. In *Modelling of Queueing Systems with Markov Chains: An Introduction to Basic and Advanced Modelling Techniques*, pages 13–88. BoD–Books on Demand, 2014.

[SYLZ00] Zhong Su, Qiang Yang, Ye Lu, and Hongjiang Zhang. Whatnext: A prediction system for web requests using n-gram sequence models. In *wise*, page 0214. IEEE, 2000.

[TC15] Marko Tkalcic and Li Chen. Personality and recommender systems. In *Recommender systems handbook*, pages 715–739. Springer, 2015.

[WIZ15] Sholom M Weiss, Nitin Indurkhya, and Tong Zhang. From textual information to numerical vectors. In *Fundamentals of predictive text mining*, pages 13–40. Springer, 2015.

[WK92] Jonathan J Webster and Chunyu Kit. Tokenization as the initial phase in nlp. In *Proceedings of the 14th conference on Computational linguistics-Volume 4*, pages 1106–1110. Association for Computational Linguistics, 1992.

[WKW+13] Gang Wang, Tristan Konolige, Christo Wilson, Xiao Wang, Haitao Zheng, and Ben Y Zhao. You are how you click: Clickstream analysis for sybil detection. In *USENIX Security Symposium*, volume 9, pages 1–008, 2013.

[ZLL+13] Zui Zhang, Hua Lin, Kun Liu, Dianshuang Wu, Guangquan Zhang, and Jie Lu. A hybrid fuzzy-based personalized recommender system for telecom products/services. *Information Sciences*, pages 117–129, 2013.