eBay Online Customer Support engagement segmentation

Edgar Cadena, B.S.

A thesis submitted to University College Dublin in part fulfilment of the requirements of the degree of Master of Science in Business Analytics

Michael Smurfit Graduate School of Business,

University College Dublin

September, 2016

Supervisor: Dr. J. Sweeney

Head of School: Professor Ciarán Ó hÓgartaigh

Dedication

To my parents for their support and encouragement

Contents

List of Figures

List of Tables

List of Algorithms

Acknowledgements

I would like to express my appreciation to Mr. Gareth James, Mr Michael Mucci and Mr. Hetal Shah from eBay's Self-service support team, and Dr James Sweeney for their valuable and insightful suggestions and support during the planning and development of this research work.

Important Abbreviations

OCS — Online Customer Support
KPI — Key Performance Indicator
PAM — Partition Around Mediods
SPADE — Sequential Pattern Discovery using Equivalence classes

Abstract

eBay's Online Customer Support team has defined a new KPI called OCS Engagement that provides information on customers' interest on using self-serving capabilities. The new KPI provides a high level view on customer self-service interactions but offers no detail around the quality or quantity of such interactions. The work presented in this Practicum aims to provide a model for extracting detailed information about self-service interactions as well as data patterns showcasing common behaviors among customers. Along with a detailed analysis, this Practicum also aims to provide eBay with a framework to derive further insights based on this work. Previous customer segmentation works based on clustering techniques were leveraged to produce this Practicum. Based on these techniques, we provide an in-depth analysis of what self-service interactions look like and how they differ among them, specifically providing data on main statistics around self-service session length and duration, which showed some surprising results (higher than expected session length and lower than expected session duration). We were also able to demonstrate that there are cluster patterns in the self-service interactions data and that they group around 10 clusters. We also characterized these clusters and provided insights in regards perceived performance from a customer experience standpoint, showing that having multiple interactions does not result in clear Resolution Rate increases. We also showed that engaged customers, those with two or more interactions are more likely to use the same self-service option rather than switching to another solution. Finally, we were able to show that all the information and new insights might be gathered from the Tableau Dashboard that was designed for this project. We believe the work

presented in this practicum is important because it sheds light into the inner workings of a newly created KPI that will be used as guidance in the design and implementation of new features expected to increase customer satisfaction and reduce call centre contacts.

Chapter 1

Executive Summary

1.1 Introduction

The eBay team in charge of managing eBay's Online Customer Support solution (OCS) has been working on defining new KPIs that would allow them to get a better, deeper understanding of how customers engage with their self-service tools. With this purpose in mind, a new KPI called OCS Engagement was defined and implemented in H1 of 2016. The definition of the new KPI is straightforward; OCS Engagement represents the proportion of customers that after reaching OCS web site's landing page interact with at least one of the available self-service offerings.

The purpose of this work is to develop a model and a framework that would allow the OCS team at eBay to delve deeper into the components that make up the OCS Engagement metric and discover patterns that might indicate common engagement behaviors. In practical terms, this works aims to answer the following questions:

- What do self-service interactions look like in terms of duration and frequency?
- Are all self-service tool interactions alike? If not, how so?

- Are there any common self-service interaction patterns in the data?
- If any patterns, can they be characterized using other useful dimensions like Site or Platform?
- Can we say anything in regards self-service interactions performance from a Customer experience standpoint?
- Are there insights that might be actionable in the short term?
- Can the OCS team derive further insights based on this work?

Data used for this project has the following characteristics:

- Timeframe: June 2016 - July 2016
- Regions: United States, Australia, Germany, United Kingdom
- OCS Session split: 70% surveyed, 30% escalated sessions
- Total OCS active sessions: 31,845

1.2 Results

1.2.1 Session length analysis

The average session length in terms of number of self-service interactions is 2.15, which is close to the number of interactions expected by the OCS team. The maximum of 67 however was surprising to the team, as such patterns are usually associated with web bots. Data show that most sessions (92%) have a session length of 4 or less while half of them have a single self-service interaction. This last result was surprising to the OCS team, which expected a higher number of interactions per session.

1.2.2 Session Duration analysis

The average session duration is 9.47 min, which is within the 10 minute range expected by the OCS team. However, although most sessions (79.57%) have a duration of 13 minutes or less about 52% percent last less than 4 minutes and almost 20% of all sessions last for less than a minute. This last result again was surprising to the team, who were expecting sessions with multiple interactions to last longer.

1.3 Key insights

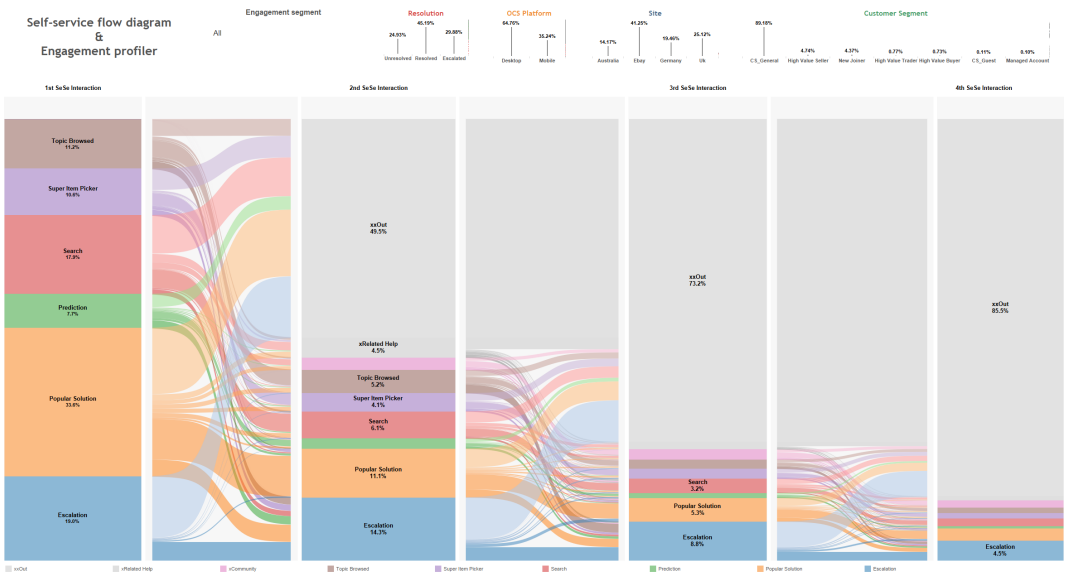

Figure 1.1: Self-service flow diagram and engagement profiler visualization

1.3.1 More interactions better resolution?

It can be shown from the dashboard that the Resolution Rate from people leaving the OCS site after the first interaction is 50.97%. This rate drops to 48.61% for customers leaving the site after their second interaction. The interesting part is that the Resolution Rate for "Lightly engaged" segment

is 50.88%, which is higher than the global average . This means that for the population that have two or more interactions their Resolution Rate in average is slightly lower than those with a single interaction. A result that was truly surprising to the team.

1.3.2 More self-service options better engagement?

Engaged customers mostly interact with the component of their preference rather than trying a different approach. It can be shown from the dashboard that "focused" clusters show a similar pattern when it comes to the choice of second component for the second interaction. That is, the repeat interaction rate with the same component. For "Search focused" this rate is 87.91% whereas "Topic Browsed focused" shows 83.69%. This result begs the question as to whether it would be better to try and optimize the experience for a single component instead of providing extra options that most engaged customer seem to ignore is most cases.

1.3.3 Proactively offer escalation options to High Value customers?

We show in Chapter 5, that the mean session length for customers with multiple interactions is about 3.2 segments per session. We also show that customer segment "High Value Sellers" is dis proportionally higher within clusters "Topic Browsed focused" and "Escalation focused". After a few discussions with the OCS team a theory emerged that perhaps "High Value Sellers" showed this particular behavior because they were mostly interested on speaking to a teammate rather than self-serving. This bit of info combined with the low mean session length and the observable drop in Resolution Rate in later segment interactions, made us wander whether the OCS team should try and optimize the experience for High Value Sellers and proactively offer an escalation channel after having 5 or more self-service interactions.

1.4 Conclusion

1.4.1 Contributions

Insights obtained with this work point out opportunities to improve the overall customer experience. Opportunities range from fast-tracking access to live channels to High Value customers that show signs of friction after surpassing a pre-defined session length threshold defined by the business team. Opportunities to influence design decisions that align with the observable desire of customers to engage with the same self-service component rather than use a different one. Another opportunity relates to optimizing OCS Platforms (desktop vs mobile) based on the clear preferences expressed by the different cluster profiles. Also an opportunity to proactively reach out to High Value customers that we know engage only once and did not get a resolution to their problem.

1.4.2 Future work

The first natural extension to this work would be to scale up the application to include other regions and more data into the analysis. The second extension would be to include site content information into the analysis. Up until now, the analysis has exclusively focused on the mechanics of the self-service interactions. Including information on the actual content presented to customers would help to come up with more refined segments and other type of insights related to content preferences. Finally, move on to complete the profiling of common paths using the Tableau visualization and cluster information.

Chapter 2

Introduction

2.1 Opening Remarks

Online customer support portals have become a necessity for any company with a large customer base in need to get prompt and accurate resolutions to their questions in a convenient and simple manner. Most of these online systems provide a variety of options for their customers to interact with curated content extracted from knowledge databases or from experiences from other customers dealing with similar queries in the past. These systems also offer an array of channels should their customers choose to bring their questions to an actual customer representative. Understanding how customers interact with these self-service tools and why and when they decide to contact the call centre instead of self-serving is of vital importance for product and service designers. Some of these insights may be incorporated into their designs with the objective of enhancing the self-service experience and ultimately increasing customer satisfaction while reducing cost from call centre contacts.

2.2 Business Context

eBay has been one of the key players in the e-commerce space for the past 20 plus years. As of late, eBay's business model has evolved from an auction-

driven marketplace into a modern e-commerce enabler where sellers and buyers alike interact in a global, highly competitive environment where customers expect nothing but the best available service. Aligned with this vision, the eBay team in charge of managing eBay's Online Customer Support solution (OCS) has been working on defining new KPIs that would allow them to get a better, deeper understanding of how customers engage with their self-service tools. With this purpose in mind, a new KPI called OCS Engagement was defined and implemented in H1 of 2016. The definition of the new KPI is straightforward; OCS Engagement represents the proportion of customers that after reaching OCS web site's landing page interact with at least one of the available self-service offerings. Figure 2.1 shows OCS' landing page of eBay's main site along with the three main self-service offerings; Search, Topic Browse and Popular Solution. Each of these self-service tools will eventually guide customers to specific content based on their preferences. Four other self-service components are available; Related Help, Super Item Picker, Community Help and Prediction. These options however are shown based on the customer's selling and buying activity and do not show up by default.

In practical terms, the OCS Engagement metric measures the level of interest that users might have on searching for content, pulling information on a popular solution or browse the site's taxonomy looking for a particular answer. That is, the level of interest in self-serving rather than directly clicking on the Contact Us button at the bottom of the page.
As useful as the new KPI is, it provides a high level view with limited insights about the actual quality of the self-service interactions. Information such as duration and number of self-service interactions are hidden behind the new metric definition. If eBay are really to get the most out of the new metric, then a deeper analysis of the components that make up the OCS Engagement metric is needed. The problem then becomes a quest for an approach that would allow the OCS team to extract detailed information about self-service interactions as well as common patterns that might showcase common behaviours or ways of interacting with self-service tools. Fortunately there are

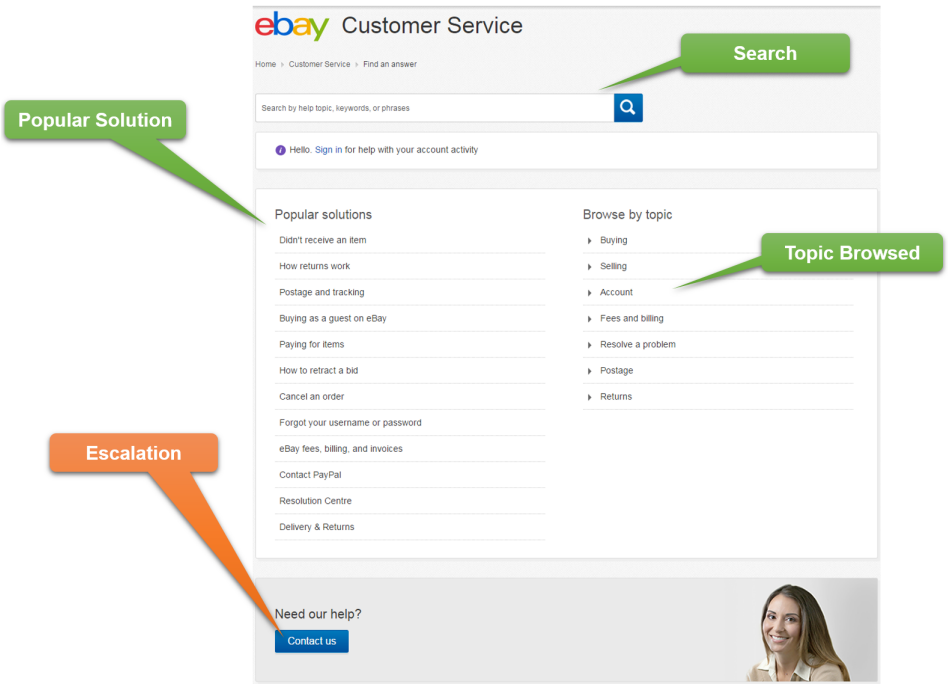

Figure 2.1: eBay's Online Customer Support landing page

plenty of examples in the Literature that tackle similar problems using different approaches and techniques. For the purpose of this work, we focused on three specific papers that provide solutions to customer segmentation problems in different contexts using data clustering techniques. Olatz *et al.* (2013) addresses the problem posted by a Tourism web site on how to segment customers based on their usage patterns and content preferences. Hung *et al.* (2013) shows a clustering analysis around self-care subfunction interactions chosen by elders on a customer care portal. Liu and Keselj (2007) also deal with a customer segmentation problem based on mining web server logs and web contents.

2.3 Business Question

This project aims to develop a model and a framework that would allow the OCS team at eBay to delve deeper into the components that make up the

OCS Engagement metric and discover patterns that might indicate common engagement behaviours. In practical terms, this works aims to answer the following questions:

- What do self-service interactions look like in terms of duration and frequency?
- Are all self-service tool interactions alike? If not, how so?
- Are there any common self-service interaction patterns in the data?
- If any patterns, can they be characterized using other useful dimensions like Site or Platform?
- Can we say anything in regards self-service interactions performance from a Customer experience standpoint?
- Are there insights that might be actionable in the short term?
- Can the OCS team derive further insights based on this work?

2.4 Practicum Outline

The practicum starts by providing a Literature review on online customer support systems, the importance of understanding customer behaviours and previous works addressing similar customer segmentation problems. In particular, we focus on three main papers that provide most of the technical guidance used in this work. We then move on to describe the methodology and techniques used to compile the data needed for the analysis as well as the procedures to align the data with the approaches in the Literature and the experiments designed to implement and test the clustering algorithms. Results from the final segmentation outcome are then discussed in the context of the business questions and a deeper analysis on key insights is presented using a visualization tool specifically designed for this project. We finish the practicum by proving our conclusions, assessment of the project and guidance on how to move forward with actionable insights and future work.

2.5 Contributions

We believe this project provides a deeper understanding of how the components of the OCS engagement metric look like and how they differ among them. It does so by providing an upfront analysis as well as a tool for the business team to conduct their own analysis and derive their own insights. We found the main statistics around frequency and duration for self-service interactions and also discovered that self-service interaction data do cluster around 10 groupings and were able to characterized the clusters using OCS' main dimensions. We also discovered a few insights that were surprising to our business partners and some others that could be actionable in the short term.

We believe the work presented in this practicum is important because it sheds light into the inner workings of a newly created KPI that will be used as guidance in the design and implementation of new features expected to increase customer satisfaction and reduce call centre contacts.

Chapter 3

Literature Review

3.1 Introduction

Given the nature of a practicum, the Literature review presented on this chapter aims to provide a review of the problem's background, what other researchers have done to solve similar problems, which of these approaches and techniques might be applied to the problem discussed in this work as well as known limitations of these approaches that might have an impact on the final outcome. In summary, the Literature review focuses on reviewing relevant work that might be leveraged for the problem at hand, as opposed to trying and identifying a gap in the Literature and a way to fill it.

We begin this chapter by providing an overview on online customer support systems and the motivation to understand customer behaviors associated with interactions with this kind of applications. We then review the current state of research around customer segmentation and discuss in detail three papers whose research theme and approaches may be leveraged by this practicum. We finish this chapter by reviewing limitations on the selected approaches and their relevance to this work.

3.2 Background and research themes

Online self-service support systems have become of great importance as a vehicle to reduce call volume (Negash *et al.* (2003)) by providing online access to companies' knowledge bases (Davenport and Klahr (1998)) and by leveraging advances in technology to meet customer's evolving self-service needs (Truel and Connelly (2013)). Negash *et al.* (2003) suggest that online support systems are as effective as the quality of the information they provide as well as the quality of the system itself. In this context, Negash *et al.* (2003) define system quality as a function of *Interactivity* and *Access*, that is "the extent to which users can participate in modifying the form and content of a media-based environment" and "the availability of the system when customers try to retrieve information, along with the ease of using the interface to contact people needed for support". In terms of this Practicum, we aim to influence the quality of the system by understanding customer self-service interactions and common behavior patterns to help drive design decisions.

Understanding common behavior patterns in OCS becomes a problem of identifying and interpreting customer segments from a self-service engagement perspective. Like Smith (1956) suggests, "Market segmentation involves viewing a heterogeneous market as a number of smaller homogeneous markets, in response to differing preferences, attributable to the desires of consumers for more precise satisfaction on their varying wants". In OCS terms, we'd like to understand what these self-service interaction preferences are in an attempt to optimize the experience in accordance with precise customer segments' self-service needs. Effective segmentation will enable the OCS team to provide differentiated service to various customer segments while at the same time allocating proper resources to each segment aligned with the team's strategy (Yao *et al.* (2014)).

The Literature includes a variety of customer segmentation examples. Hamka *et al.* (2014) discuss an approach to customer segmentation based on smartphone measurement while Liu and Keselj (2007) present a customer segmentation study combining web logs and web contents mining. There are also several examples of different segmentation approaches like the one based on

rough sets suggested by Dhandayudam and Krishnamurthi (2014), support vector machines as described by Albuquerque *et al.* (2015) or a clustering approach using k-means implemented by Olatz *et al.* (2013). To our knowledge, however, there are no examples in the Literature of customer segmentation problems applied to online support systems. For that reason, we focused our search on papers similar in nature to our use case, that is, we searched for papers that describe customer segments in terms of usage or selection of distinct options in an online or web context. We found three papers that meet the above criteria. Olatz *et al.* (2013) addresses the problem posted by a Tourism web site on how to segment customers based on their web usage patterns and site content preferences. Hung *et al.* (2013) show a customer segment analysis around self-care sub function interactions chosen by elders on a customer care portal. Liu and Keselj (2007) also deal with a customer segmentation problem based on mining web server logs and web contents.
The following section describes in detail each of the papers selected as the technical basis for this Practicum.

3.3 Selected approaches

The first paper that we found was the one by Olatz *et al.* (2013). In this work, the authors present an approach to customer segmentation based on mining of web usage log files and the content of the web site. We found this work relevant because it models page visits to the Tourism site as sequence of events that could relate to the usage of self-service components in the OCS case. Olatz *et al.* (2013) describe two distinct clustering operations. The first one is based on web usage patterns extracted from log files and its main purpose is to generate user navigation profiles to be used for link prediction. Authors generate user navigation profiles by first transforming web session information into sequences of URLs visited by customers. This information is used by a clustering algorithm known as PAM (Partitioning Around Medoids) (Kaufman and Rousseeuw (2009)) which uses the Edit Distance metric (Gusfield (1997)) as the similarity metric. The result from this experiment is k clusters that

represent users with similar navigation behaviors. Authors take this result a step further and apply SPADE (Sequential Pattern Discovery using Equivalence classes) to extract common click sequences on each of the clusters. The second clustering operation on this paper was designed to obtain user interest profiles. In this case, authors start by extracting topic information from each of the URLs visited by the users with the purpose of understanding what type of content users were interested in visiting. By looking at all URLs visited in a session, authors were able to derive the *interest* users had in different topics, in other words, session information is transformed into a vector of preferences, with each dimension representing the level of interest in a particular topic. The new set of vectors of preferences are run through a k-means clustering algorithm in order to find the user interest profiles. As discussed, we found this paper encouraging since both of the clustering use cases could be modeled using self-service interactions instead of page visits. Moreover, we believe the high level description of the algorithms and testing procedures could be as well leveraged. However, we found information on actual parameters missing, a glaring example is the lack of information on what particular k-means method was used or what procedure was followed to determine k. Overall, we see this work a step in the right direction as it includes key concepts and techniques that might be leveraged in this work.

The second paper that we reviewed in detail was the one produced by Hung *et al.* (2013). In this work, authors address the problem of understanding the behavior of elderly people interacting with a online system that provides access to self-care activities or sub-functions. We found this work relevant because it also models access to different self-care functions as a sequence of events and it also looks at these interactions in terms of level of interest shown by users while interacting with the system. Just as in the case of Olatz *et al.* (2013), Hung *et al.* (2013) start by transforming web usage information into data representations more conductive for the data mining analysis in turn. Hung *et al.* (2013) suggests a sequence-based representation based on a transition matrix of a Markov chain where a state is defined as a self-care sub-function. The second type of representation suggested by Hung *et al.* (2013) is called

interest-based representation. In this case, each session in the self-care portal is transformed into an m-dimensional vector where each dimension represents the level of interest in a particular sub-function in the self-care portal. Interest in a particular sub-function is determined as a function of two main factors: "Frequency" and "Duration". "Frequency" represents the relative number of times that an elder person chose to interact with a particular sub-function. "Duration" represents a relative measure of the time spent on each one of the sub-functions in a given session. Using these two indicators Hung *et al.* (2013) provide an equation that calculates the level of interest on every available sub-function to the self-care user. Authors then describe the clustering analysis performed on both representations. First, they look at a Neural Network method (ART2) to identify the correct number of clusters in the data. After k was determined, k-means was applied to both session representations and a set of clusters was identified for each of them.
We found this paper to be the closest to the OCS self-service use case that this work aims to solve. We believe most of the concepts and techniques used in this paper could directly be applied after small tweaks. The paper includes enough information on each of the experiments so that they can be replicated. Perhaps the only critique is the explanation of the Interest equation and how it is derived, although to their credit, authors did call out that the entire procedure was entirely based on a different paper. All in all a great paper that will be used as the primary guidance source throughout this Practicum.

The last paper that we reviewed in detail was the one created by Liu and Keselj (2007). The reason we wanted to review this paper is that it is the original paper where the concepts of Frequency, Duration and Interest are introduced. In their work, Liu and Keselj (2007) use web logs and web contents data to create user navigation profiles and to predict future requests. Authors accomplish this by mining web usage data and creating a session interest-based representation similar to the one described in Hung *et al.* (2013). Most of the concepts described by Liu and Keselj (2007) are included in Hung *et al.* (2013). The main difference that we found was the explanation as to why to use the harmonic mean of "Frequency" and "Duration" to calculate "Interest" which

has to do with a standard formula that is used to calculate the harmonic mean of two ratios. Apart from this bit of information, the rest of the paper is very similar to Hung *et al.* (2013). Therefore, we decided to stick with Hung *et al.* (2013) as the primary paper and use Liu and Keselj (2007) as a reference.

3.4 Limitations

Based on the papers selected as the basis for this project we see the following limitations on the techniques proposed for this work:

- k-means algorithm: Need to provide k parameter. This means using a secondary method to discover k before applying the actual clustering algorithm. Although a limitation, not much of a concern since at least two of the three papers discuss how to estimate k

Chapter 4

Methodology

An experimental methodology was developed to address the business question posed in the Introduction chapter. This chapter describes in detail this methodology, including a description of the experimental environment as well as the procedures used during the design, development and testing of the analytic model.

4.1 Experimental Environment

All experiments for this practicum were conducted using eBay's hardware and software infrastructure. This section provides information on specific hardware and software so that another researcher could reproduce the results presented in following sections.

4.1.1 Hardware

With the exception of the database server, all other procedures related to the experiments presented in this section were performed using eBay's standard development hardware with the following specifications:

- Processor: Intel Core i7-5600U @ 2.60GHz

- RAM: 8.00 GB
- HDD: 475 GB

4.1.2 Software

As discussed in the literature review, several clustering approaches have been successfully used in customer segmentation problems. As this work is based on such approaches, it was necessary to get access to software with the desired capabilities while making sure it was available within eBay's infrastructure. Given these constraints, we decided to go with the following software:

1. R (version 3.2.2 64-bit). R is one two main platforms for data analysis widely available at eBay. It is well supported and there's an active internal community working with it. Also, R provides access to packages that cover all the algorithms that will be used in this work. R will be primarily used for data extraction, data preparation and to develop and test the clustering solution.

2. Tableau (version 9.1 64-bit). Tableau is the experimental and prototyping platform of choice at eBay. It is widely used and accepted by both data analysts and end-users who have become used to Tableau's capabilities and ease-of-use. Tableau will be used as the main front-end for this project. It will provide access to experiment results as well as visualizations that highlight key findings.

3. Teradata SQL assistant (version 14.10.0704). Although most of the heavy lifting was done in R, the initial data analysis and discovery was done running queries and analyzing results within Teradata SQL assistant.

Although the initial set up for this project was enough to run the experiments and validate the results, it will have to be scaled up in terms of memory and CPU in order to analyze larger datasets. More information and a recommendation on how to achieve this will be presented in the Conclusion and Future work section.

4.2 Model Development

We approached the development of the solution as a regular data analytic project. This section describes in detail the work done at all different phases of the knowledge discovery and data mining process.

4.2.1 Data Acquisition

As stated in the Introduction section, our goal is to analyze usage patterns of eBay's customers interacting with eBay's online self-service components. eBay stores all web usage information, including online self-service interactions, in a large database using a proprietary format. Information specific to OCS interactions is extracted, transformed and loaded into a set of tables designed to specifically model the journey that members take while interacting with eBay's OCS as shown in Figure 4.1.

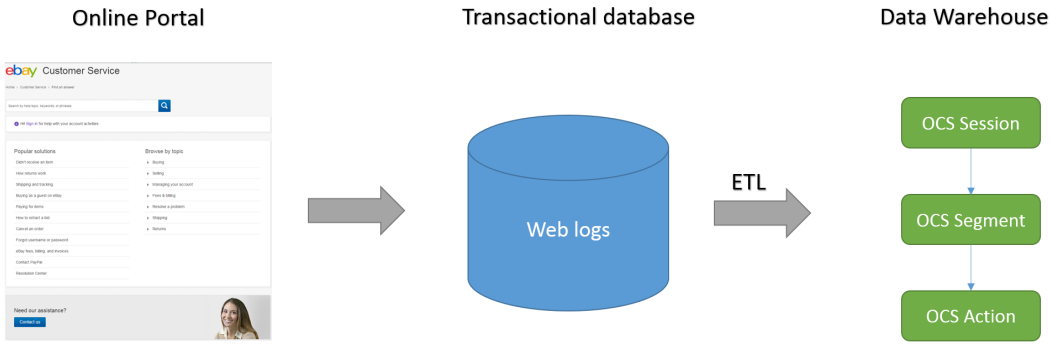

Figure 4.1: OCS' data extraction and transformation process

This set of three tables represents the main data source for this project. OCSSession table includes data related to the overall experience including session date, session duration, user ID, customer segment and other high level information. OCSSegment table stores information on the different interactions that customers had with actual self-service components, e.g. if a user performs a search looking for information on how to reset her password, then such operation is logged as a self-service segment on this table. OCSSegment table also

includes information on the content that was reviewed and the order on which such interactions occurred. Finally, OCSAction table stores granular information in the form of pair-value attributes that represent specific parameters or actions taken by customers while interacting with a particular self-service solution. This level of detail is required to differentiate between very particular interactions and to accurately measure the time spent on each of them. The main task from a data acquisition perspective was to create a SQL query capable of combining information from all three tables so that it can be used in the data processing phase. This SQL was then enriched to include other related customer information such as tenure and customer segment which are needed for profiling purposes. Table 4.1 shows a sample set of records combining data from all three tables.

site	session_id	session_begin_date	customer_segment	attribute_id	attribute_value
Ebay	5423377310	15/07/2016 18:54:37	High Value Trader	500000441222	3104
Ebay	5423377310	15/07/2016 18:54:37	High Value Trader	5000002470	3104
Ebay	5423377310	15/07/2016 18:54:37	High Value Trader	5000002910	OcsSelfService

Table 4.1: Sample OCS data combining session, segment and action data

Once a SQL solution was found the next step was to decide on the actual dataset to pull from the data warehouse. Recall from the Introduction chapter that one of the main objectives of this analysis is to understand the performance of self-service interactions from a customer perspective. To that end, we wanted to pull all OCS session information from customers that were surveyed and sent a response back. Also, a tracking bug that was fixed in May 2016 would limit us to pull data from June 2016 forward. With all these constrains in mind the final dataset agreed with the business partners has the following characteristics:

- Timeframe: June 2016 - July 2016
- Regions: United States, Australia, Germany, United Kingdom
- OCS Session split: 70% surveyed, 30% escalated sessions

The 70/30 split of sessions corresponds with the actual escalation rate observed at eBay. That is, in general, 30% of all sessions started in the OCS portal are escalated to the call centre and handled by a customer support representative. In order to achieve the 70/30 split we decided to pull all OCS surveyed sessions (about 40K sessions) and sample 17K OCS sessions that ended up being escalated and handled by a customer support representative.

4.2.2 Data Preparation

It is interesting to note that even though eBay's OCS data model had been designed to facilitate analysis on self-service metrics, its current pair-value attribute representation is not ideal for the analysis proposed in this work. In order to follow the segmentation approach suggested by Hung *et al.* (2013) and by Olatz *et al.* (2013) data collected in the previous step had to be transformed into the sequence-based and interest-based representations suggested in both papers.

Sequence-based representation

As its name suggests, a sequence-based representation is constructed by listing or *sequencing* elements that occur in a given order. In the case of Olatz *et al.* (2013) a sequence represents URLs visited by customers in a single session whereas in Hung *et al.* (2013) sequences represent interactions with sub-options in the self-care customer portal. In our case, sequences would represent interactions with OCS self-service components listed in chronological order. Figure 4.2 describes an OCS session in which a customer engaged with three self-service components. Each of these three interactions is then represented as a segment in a sequence of events. In this sample case the total sequence length is 3.

In practical terms, sequencing is achieved by using an R script that traverses the list of pair-value attributes per session extracting information on what self-service components were engaged and the amount of time that the customer

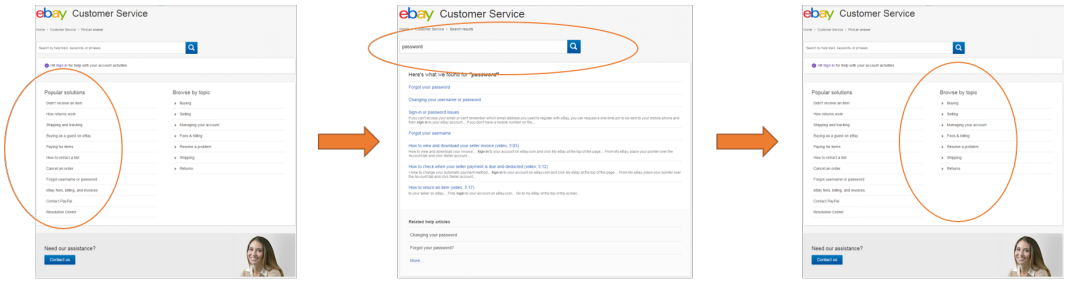

Figure 4.2: OCS interactions represented as a sequence of segments

spent with each of them. The final sequence representation is a string of characters where every letter is mapped out to a self-service option, e.g. Topic Browsed interactions are mapped to T, Search interactions to S and so on for the rest of them. This simplified representation will be useful when calculating similarity distances among sequence strings. Figure 4.3 shows four sample sessions, their three first segments and their simplified sequence string.

```
> seeded[51:54,c(10:12,9)]
           seseSegments.1.          seseSegments.2.  seseSegments.3. seqString
51         (CU)-Escalation                                                   E
52         (CU)-Escalation (SeSe)-Popular Solution                          EP
53 (SeSe)-Popular Solution    (SeSe)-Topic Browsed (SeSe)-Community    PTCCSSE
54         (CU)-Escalation                                                   E
```

Figure 4.3: OCS sessions represented as sequence of segments in R

After processing all sessions, every single session will be represented as a vector of characters representing a session sequence. Let O be the set of OCS components visited by a user in a session, $O = \{o_1, o_2, ..., o_m\}$, where m is number of components visited in a single session and $o_i \in \{T, S, P, H, E, I, C, R\}$ one of 8 possible OCS components. Let S be a set of user OCS sessions. Hence $S = \{s_1, s_2, ..., s_n\}$, where n is the total number of sessions and s_i is a subset of O.

Interest-based representation

Interest-based representations are useful to represent intent or degree of interest among a variety of options. That is, given a set of available options, whether different content in a tourism web site (Olatz *et al.* (2013)) or sub-functions used by elders in a self-care portal (Hung *et al.* (2013)), an interest-based representation assigns a number between 0 and 1 to each option which represents the degree of interest in that option by a particular customer.
In the work by Olatz *et al.* (2013), a set of vectors representing customer interest in different content of the tourism site were built with this purpose. In Hung *et al.* (2013) the interest-based representation was derived as a function of the frequency and duration of sub-functions used by elders within a single session. It is actually the approach suggested by Hung *et al.* (2013) which we found closer to the use case that this work aims to solve. The intent-based representation implemented in this thesis is based on the ones presented by Hung *et al.* (2013) and Liu and Keselj (2007). In both cases two measures are used to determine a user's interest in a particular option: *Frequency* and *Duration*.
In the context of this work, *Frequency* is defined as the number of uses of an OCS self-service component relative to the total number of uses of OCS self-service components in a session. The underlying assumption is that self-service components with higher frequency are of more interest or useful to eBay's customers. Let C be the set of OCS components available to OCS users, $C = \{c_1, c_2, ..., c_8\}$ Then *Frequency* is given by Eq 4.2.1

$$Frequency(c_i) = \frac{NumberOfUses(c_i)}{\sum_{c_i \in C}(NumberOfUses(c_i))} \tag{4.2.1}$$

Duration is defined as the total time spent interacting with an OCS self-service component within a session. In other words, the elapsed time between one OCS self-service interaction and the next one aggregated by components of the same type. Again, the assumption is that longer durations reflect greater interest

in a particular component. This assumption requires the expected interaction time to be the same across components. However, it is possible that some interactions are naturally longer than others (Liu and Keselj (2007)), e.g. searching operations could take longer in average than clicking on a pre-selected Popular Solution. To alleviate this problem, total duration per component on a session is divided by its average duration. Average component durations were calculated as a byproduct of the work done sequencing OCS self-service interactions. One final consideration is last segment durations (Liu and Keselj (2007)). In our case the duration assigned to the last interaction is the average duration across components in the session as suggested by Liu and Keselj (2007). To finalize, *Duration* is further normalized by the max *Duration* in the session as shown by Eq 4.2.2

$$Duration(c_i) = \frac{TotalDuration(c_i)/AvgDuration(c_i)}{max_{c_i \in C}(TotalDuration(c_i)/AvgDuration(c_i))} \tag{4.2.2}$$

In the same case as in Liu and Keselj (2007), both *Frequency* and *Duration* are considered strong indicators of customers' interest. Moreover, both indicators are given the same importance. For this reason and considering we'd like to measure the average of two rates it would then be appropriate to use the harmonic mean of *Frequency* and *Duration*, as suggested by Liu and Keselj (2007), as a way to measure the degree of interest in an OCS component by a customer. Eq 4.2.3 shows the final proposed formula of *Interest*:

$$Interest(c_i) = \frac{2 \times Frequency(c_i) \times Duration(c_i)}{Frequency(c_i) + Duration(c_i)} \tag{4.2.3}$$

In terms of implementation, *Frequency*, *Duration* and *Interest* measures are calculated on-the-fly as session data is transformed into a sequence based rep-

resentation. At the end of session processing, each session is converted into an 8-dimensional vector of interests of OCS components, i.e. $s = \{i_1, i_2, ..., i_8\}$. This final set of session-interest vectors will be used as input to the clustering algorithms in the following sections.

As a final note in the Data preparation section, once all OCS sessions were transformed, we took a closer look at the final dataset and removed sessions with no segments. These data inconsistencies are primarily due to tracking issues across different OCS components. The final dataset was then reduced to 31,845 sessions in total.

4.2.3 OCS session clustering

Once sequence-based and interest-based representations of OCS sessions have been constructed, the next step in the analysis is to find groups of sessions that share similar behaviours. As indicated in the Literature review, unsupervised algorithms such as clustering have proven to be successful while dealing with these type of problems. This section describes the clustering approaches followed by Olatz *et al.* (2013) and Hung *et al.* (2013) and how they were leveraged in the context of this work.

Interest-based session clustering

For clustering of interest-based representations the authors of both Hung *et al.* (2013) and Liu and Keselj (2007) coincide in using k-means as the main clustering algorithm. They also agree on taking the Euclidean distance as the similarity or distance measure. However, they differ on the method to identify the optimal number of clusters. In the case of Hung *et al.* (2013) a Neural Network approach was used to determine the number of clusters. On the other hand, Liu and Keselj (2007) opted for an approach using internal evaluation functions known as cluster compactness and cluster separation. Although both approaches seemed to be adequate to finding k a third approach using Hierarchical Clustering was proposed by a colleague who had experience using this algorithm in R. Another appealing aspect to using Hierarchical Clustering is

the ability to compare multiple clustering methods outputs as a way to validate or evaluate clustering performance. Hence we decided on using Hierarchical clustering as the method to determine the number of clusters.

The method used to determine k starts by creating a distance matrix using the Euclidean distance measure. This distance matrix is then used as input to the Hierarchical Clustering algorithm, which is called with default parameters. We then plot a histogram of distances in the similarity matrix and a line graph of distances traveled (elbow chart). These two visualizations should show special features in case the data are grouped around clusters Pang-Ning *et al.* (2006). The code snipet below shows the basic R code used to determine k.

```
distSeSe <- dist(seseSample[,c(76:83)],method="euclidean")
hcSeSe <- hclust(distSeSe, method="ward.D")
memberHC <- cutree(hcSeSe, k=k)
layout(matrix(c(1,1,2,2), 2, 2, byrow = FALSE))
hist(distSeSe,col = "gray", breaks=100)
plot(hcSeSe$height[n:(n-50)],type="b",)
rect.hclust(hcSeSe,k=10)
```

Algorithm 1: Set of R commands to determine number of clusters k

This procedure was executed 10 times as a start to get a feel of the possible number of clusters using default parameters. Figure 4.4 shows a sample output from this procedure.

A few interesting features can be observed in Figure 4.4. The histogram of distance matrix shows 10-12 peaks that may be attributed to the same number of clusters. The chart of distances traveled shows a bend or *elbow* right at the 10 marker, thus suggesting possibly 10 clusters in the data. Although not conclusive, these results provided us with enough evidence to carry on a larger experiment involving both Hierarchical Clustering and k-means methods as well as a variety of parameters that were tested all at once.
The heart of the experiment consists in comparing the output of both clustering methods while varying several input parameters. The motivation for this

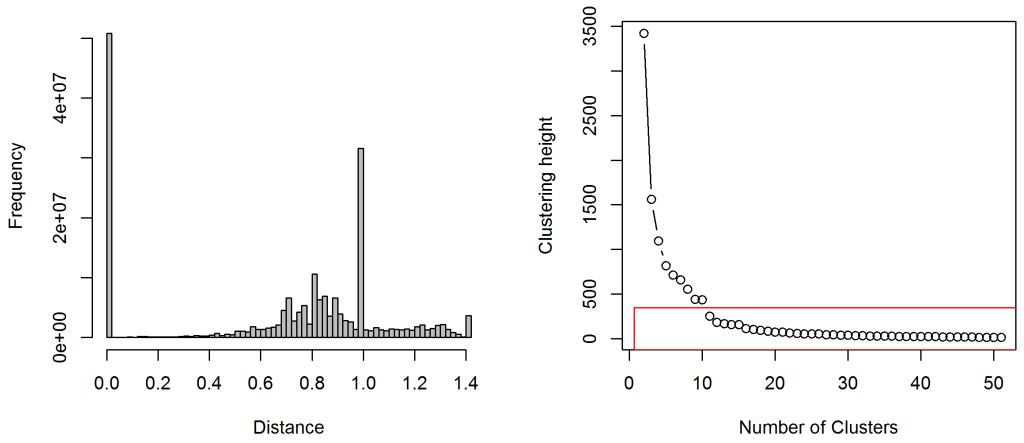

Figure 4.4: Visual aids to determine k, number of Interest-based clusters

experiment is to try and find the best possible parameter configuration, i.e. the one with the best performance. The performance metric chosen for the experiment is the ratio of sessions that are clustered outside the same cluster by both clustering methods, i.e. the cluster membership error across clusters. The main parameters to vary are the method used by Hierarchical Clustering and the method to be applied by the k-means algorithm. We carried out the experiment a total of 30 times and the results are summarized in Figure 4.5

Each boxplot in Figure 4.5 shows the distribution of cluster membership errors for a given combination of parameters after 30 iterations. For example, the mean ratio for the first boxplot is 3.24%, which means that in average there were 3.24% sessions clustered in different clusters while using the "ward.D" method for hierarchical clustering and the "Lloyd" method for the k-means algorithm. Two main observations can be made from Figure 4.5. Notice how the boxplot with the lowest mean is the combination "ward.D2" and "Lloyd". Moreover, it is also the only combination with no outliers and the lowest variance. However, we don't believe there's enough evidence to select a particular k-means method over the others but we can at least conclude that performance in terms of cluster membership error does not seem to vary much across k-means method. With this information we decided to go with the "Lloyd"

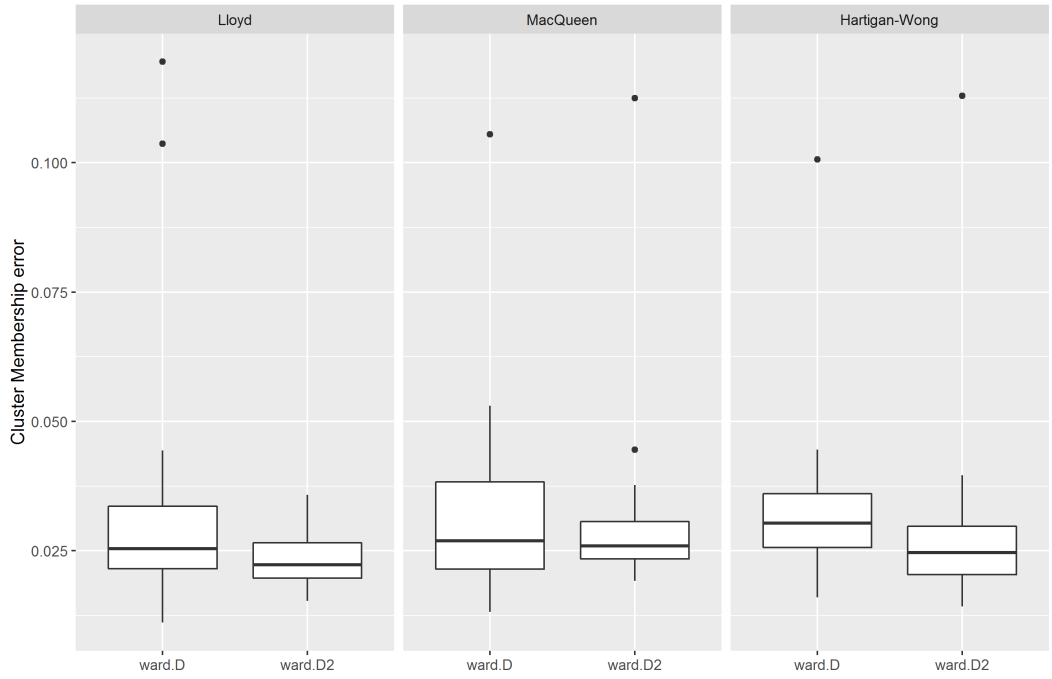

Figure 4.5: Box plots of cluster membership errors using different parameters

method since it does not seem to make any significant difference in the end. The main statistics for this set of parameters is shown in Table 4.2

Min	Median	Mean	Max
1.53%	2.23%	2.35%	3.58%

Table 4.2: "ward.D2" and "Lloyd" cluster membership error statistics

We believe the information provided by these experiments allow us to conclude that for this system, it is resonable to use the k-means algorithm with the "Lloyd" method and expect in average good performance for a $k = 10$ parameter. The final clustering operation was then carried out with the suggested parameter configuration and the results will be presented in the next chapter.

Sequence-based session clustering

As with the case of interest-based clustering, we look again at the main two papers this work is based on for suggestions on how to proceed with sequence-based clustering. In this case, we decided to follow the approach in Olatz *et al.* (2013) which suggests the use of the Edit Distance sequence alignment method (Gusfield (1997)) as the metric to compare string sequences. Using this metric is convenient within the R environment as there are multiple libraries capable of calculating the Edit Distance metric and return a distance matrix object like the one used in the previous section. Although not mention in the paper, we believe this might be one of the reasons Olatz *et al.* (2013) decided to use PAM (Partitioning Around Medoids) (Kaufman and Rousseeuw (2009)) as the clustering algorithm since it can work with either a distance or dissimilarity matrix. Using PAM also offered the alternative to use an approach like the one discussed in the previous section where we had the chance to compare two clustering methods. In this case, the experiment would be set up to compare Hierarchical Clustering and PAM across a variety of parameters.
We start again by estimating k using the same methodology as in the previous section. That is, we run the Hierarchical clustering algorithm for 10 iterations and visually inspect the resulting charts. The results from this experiment are shown in Figure 4.6.

As encouraging as the previous results were the results from this experiment are not as clean-cut as expected. The distance traveled chart shows where the line would cut if there were 10 clusters. However, just by looking at the top of chart we could make the case that there are only 5 clusters or 15 if focusing at the bottom of the chart. In other words, there is not as clear evidence as the actual number of clusters like in the previous section. However, since a similar experiment was already set up we decided to make k one of the varying parameters along with the method used to calculate the Edit Distance Metric. All of this while keeping fixed the Hierarchical Clustering method to "ward.D2". The experiment results are summarized in Figure 4.7.

The first thing to notice in Figure 4.7 is the greater variance in the cluster membership error across k. For $k = 5$ the error is as high as almost 30% in

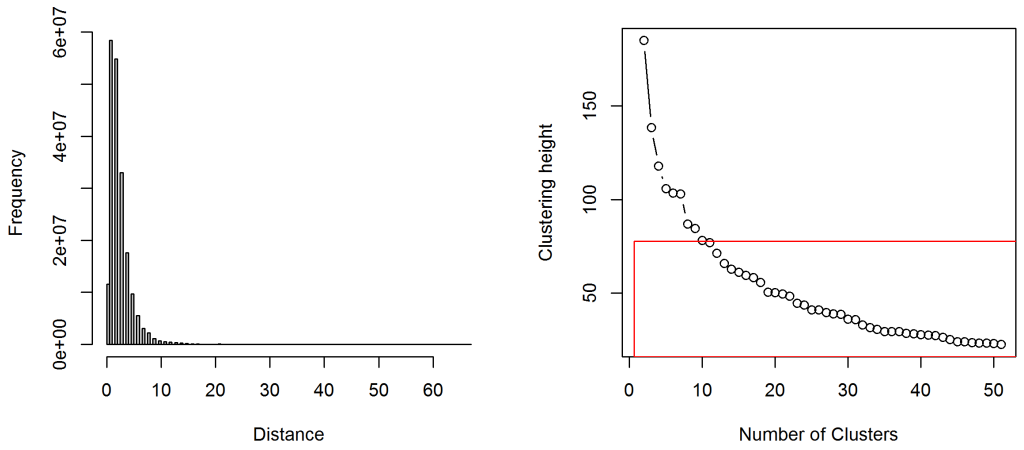

Figure 4.6: Visual aids to determine k, number of Sequence-based clusters

some cases. However the mean and variance measures of the cluster membership error are lower for higher values of k, although still higher than the errors reported in the interest-based experiment. Compare for instance the best performance results from the combination $k = 15$ and the "osa" method shown in Table 4.3 with the best results from the interest-based experiment.

Min	Median	Mean	Max
3.52%	7.41%	7.45%	14.43%

Table 4.3: $k = 15$ and "osa" method cluster membership error statistics

In the PAM case, all error statistics are higher than its k-means counterpart. Even with these results we still decided to go ahead and use the best possible parameter combination and run the PAM algorithm against the main data set and have a second look at the resulting profiling analysis to determine whether the clusters obtained with PAM are meaningful in a business context.

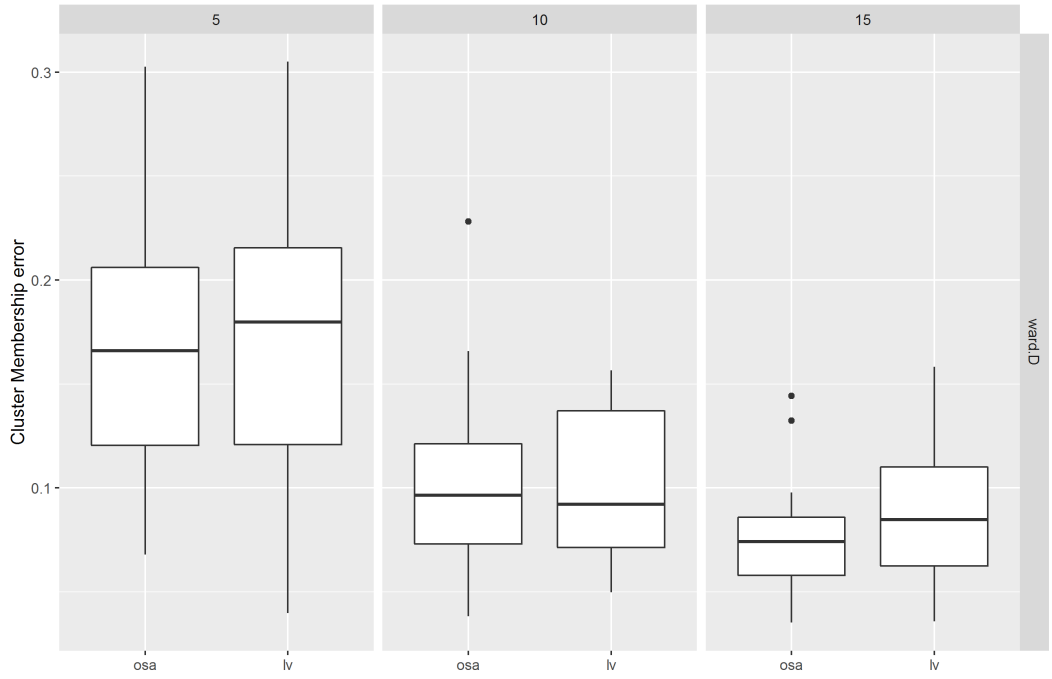

Figure 4.7: Box plots of cluster membership errors using different parameters

Chapter 5

Results

5.1 Introduction

We begin this chapter by proving a description of the attributes and measures that make up the dataset obtained in the last chapter. We then move on to describe and characterize the new set of clusters as well as profiling them against the self-service session attributes in the dataset.
We'd like to note that all of the charts and results presented in this chapter were captured from a Tableau Dashboard that was built as the main deliverable for the OCS team at eBay. Apart form including information on the new clusters and key findings, the dashboard was designed to highlight the main futures of the clusters so that end-users may derive their own insights.

5.2 Profiling attributes

Recall from Chapter 4 that our final dataset includes 31,845 sessions from the original 45K. These sessions are distributed across 4 dimensions; Site, Session Outcome, Customer Segment and OCS Platform, as shown in Figure 5.1. Distributions are aligned with the actual distributions observed during the same time period and as expected, most sessions were generated on the main eBay site by customers in the general population and using mostly OCS' desktop

version.

As discussed in the Introduction, session outcome information is of particular importance as the analysis intends to provide an idea of cluster performance by profiling the clusters against the actual session outcome, i.e. whether customers indicated that their issue was resolved, unresolved or an actual escalation to the call centre happened.

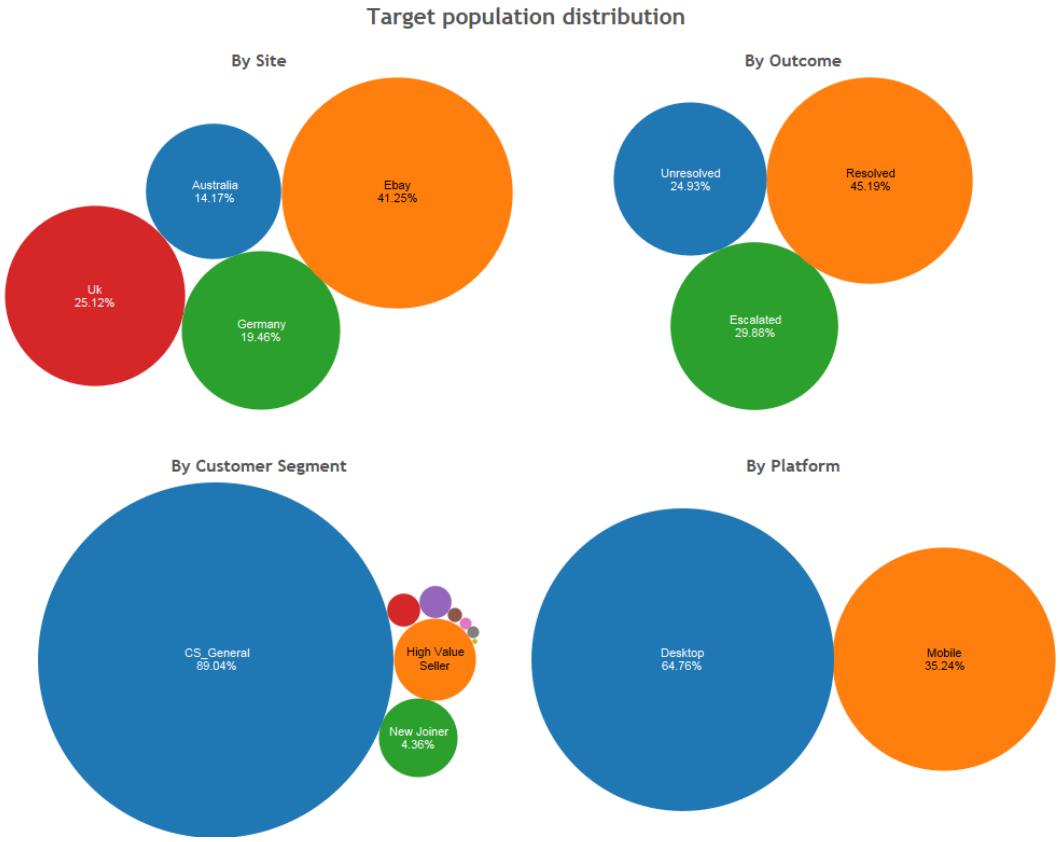

Figure 5.1: Profiling attributes and their distributions in the final dataset

5.3 Self-service interactions analysis

As suggested by Hung *et al.* (2013), this work also aims to provide insights about session length in terms of number of self-service interactions as well as session duration across all interactions. This would serve the double purpose of

providing quantitative information about self-service interactions to the OCS team as well profiling information to compare and contrast the clusters.

5.3.1 Session length analysis

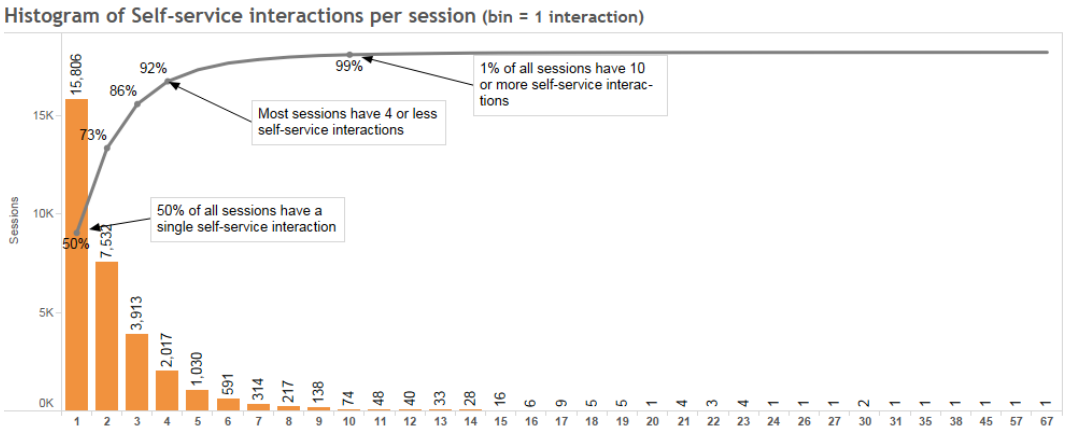

Figure 5.2: Histogram of interactions per session or session length

Figure 5.2 shows the distribution of session lengths (self-service interactions per session). The mean of the distribution is 2.15, which is close to the number of interactions expected by the OCS team. The maximum of 67 however did surprise the team, as such patterns are usually associated with web bots. The data show that most sessions (92%) have a session length of 4 or less while half of them have a single self-service interaction. This last result was surprising to the OCS team, which expected a higher number of interactions per session. The result however is not necessarily a bad result, as long as it can be confirmed that these users got a resolution to their problem. Finally, we observed that about 1% of the sessions have 10 or more self-service interactions, which seems a small percentage but it might be valuable to understand what might be causing these edge cases.

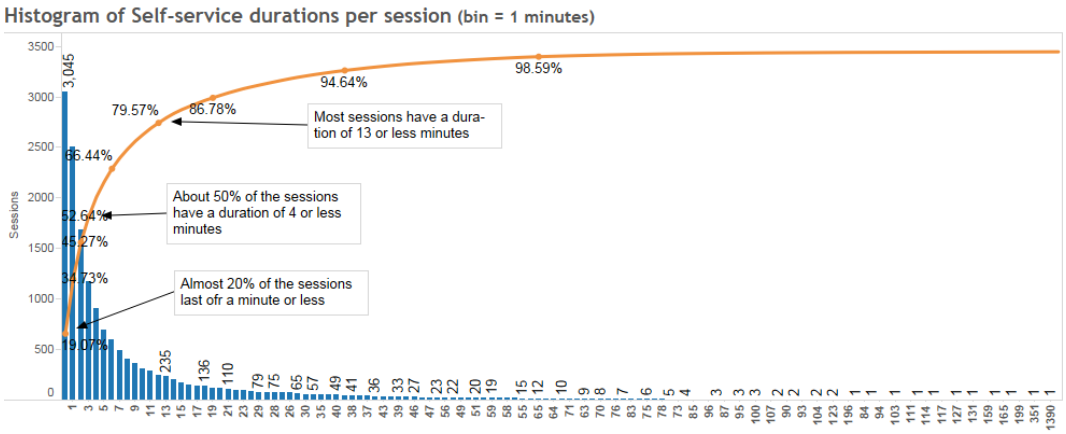

Figure 5.3: Histogram of interactions per session or session length

5.3.2 Session Duration analysis

Figure 5.3 shows the distribution of session duration in minutes. This chart only includes sessions with 2 or more interactions. This is because sessions with a single interaction would have a total duration of zero due to way we are calculating last-segment durations. The mean of the distribution is 9.47 min, which is within the 10 minute range expected by the OCS team. Looking closer though, we noticed that although most sessions (79.57%) have a duration of 13 minutes or less about 52% percent last less than 4 minutes and almost 20% of all sessions last for less than a minute. This makes us believe that the median of 3.6 minutes is a better reflection of central tendency for the session duration distribution. This last result again was surprising to the team, who were expecting sessions with multiple interactions to last longer.

5.4 Interest-based cluster results

5.4.1 Interest-based cluster patterns

We now move on to present the results from the clustering procedure described in Chapter 4. As previously discussed, the k-means algorithm cluster the data around the 10 clusters suggested by the Hierarchical Clustering method. Figure

5.4 shows all 10 clusters as rows characterized by the 8 interest dimensions as columns.

Interest-based cluster	Avg. Community Interest	Avg. Escalation Interest	Avg. Item Picker Interest	Avg. Popular Solution Interest	Avg. Prediction Interest	Avg. Related Help Interest	Avg. Search Interest	Avg. Topic Browsed Interest
Popular Solution focused	0.48%	**42.20%**	0.91%	**67.98%**	0.54%	0.37%	0.54%	1.02%
Topic Browsed focused	0.91%	14.93%	1.73%	7.01%	0.53%	0.96%	3.15%	**74.16%**
Prediction focused	0.59%	11.77%	4.20%	6.37%	**70.25%**	2.78%	2.41%	2.09%
Search focused	1.31%	8.70%	0.87%	7.29%	0.35%	2.58%	**82.13%**	1.24%
Lightly engaged	0.04%	0.10%	0.05%	0.05%	0.03%	0.05%	0.04%	0.05%
Popular Solution proficient	0.53%	0.64%	1.54%	**96.03%**	0.54%	0.28%	1.30%	0.76%
Escalation focused	0.15%	**91.90%**	1.41%	2.52%	1.65%	0.42%	1.21%	2.55%
Related Help focused	0.44%	3.34%	4.52%	10.33%	0.82%	**59.37%**	21.28%	6.69%
Super Item Picker focused	0.35%	9.07%	**77.16%**	11.93%	0.89%	1.09%	1.92%	0.98%
Community focused	**63.84%**	1.92%	1.95%	7.93%	1.84%	2.95%	16.36%	4.73%

Figure 5.4: Interest-based clusters profiling information

Looking at the mean interest across self-service components a few interesting patterns emerge. Nine out of ten clusters show a clear preference for interacting with a particular self-service component, with cluster 6 "Popular Solution proficient" showing the highest component interest (96.03%) across all clusters. Other interesting fact about cluster "Popular Solution proficient" is that it shows no interest greater than 2% for any other component. We decided to name this cluster as "Popular Solution proficient" because , as later will be shown, it is the cluster with the highest resolution rate. The reminder 8 clusters with a clear preference show highest interests ranging from 63.84% to 91.9%. Each one of these clusters was named using the name of their domi-

nant self-service component followed by the word "focused". Cluster number 10 "Lightly engaged" represents the special case of OCS sessions that did not show any perceived interest in a particular component. This can be explained by looking at the mean session length and the mean session duration, which are 1.04 and 0.16 respectively in Figure 5.5. In other words, these are the set of users that chose to interact only once. For that reason we decided to name this cluster as "Lightly engaged", because it clearly shows the lowest level of engagement, which again might not be a bad situation as long as most of these users got a resolution to their issue. Another interesting observation is the highest mean session length which corresponds to the "Community focused" cluster in Figure 5.5. We believe this is due to the fact that users are forced to take an extra step in order to be presented with any community interactions. On the other hand, the highest mean duration goes to cluster "Escalation focused". This last result is concerning as it signals at users spending a considerably longer amount of time on the site without getting a resolution and having to escalate their issues to the call centre. Continuing with the analysis, we noticed that after removing the clusters with the highest and lowest mean session length the rest of the clusters show a mean session length of 3.2. We believe this number better represents the actual behavior of users with multiple self-service interactions.

5.4.2 Interest-based cluster profiling

In this section we will show how each interest-based cluster is distributed across every one of the profiling attributes chosen for this analysis. We start by showing the OCS session outcome distribution by interest-based cluster as shown in Figure 5.6.

As previously noted, cluster "Popular Solution proficient" shows the highest percentage of resolved sessions at 66.05%, with "Community focused" coming in second. A notable mention is cluster "Lightly engaged" whose resolution rate at 50.88% is slightly higher than the 45.19% global average shown in section "Profiling attributes". The worst performers from a resolution perspective are "Topic Browsed focused" and "Popular Solution focused" and curiously

	Session metrics		
Interest-based cluster	Avg Segments Session	Avg Duration Session	% of Sessions
Popular Solution focused	3.22	8.21	5.8%
Topic Browsed focused	3.23	7.06	6.7%
Prediction focused	2.98	9.43	4.6%
Search focused	3.14	9.94	5.9%
Lightly engaged	1.04	0.16	50.1%
Popular Solution proficient	3.15	7.84	6.3%
Escalation focused	3.39	18.02	5.8%
Related Help focused	3.01	7.50	5.5%
Super Item Picker focused	3.24	8.21	5.6%
Community focused	4.41	7.99	3.7%

Figure 5.5: Interest-based clusters profiling information

	Popular Solution focused	Topic Browsed focused	Prediction focused	Search focused	Lightly engaged	Popular Solution proficient	Escalation focused	Related Help focused	Super Item Picker focused	Community focused
Unresolved	15.80%	22.39%	22.91%	29.60%	24.82%	32.50%	6.49%	34.79%	26.85%	38.60%
Resolved	20.02%	29.59%	38.61%	48.82%	50.88%	66.05%	3.60%	50.48%	49.52%	53.13%
Escalated	64.18%	48.01%	38.48%	21.57%	24.31%	1.45%	89.91%	14.73%	23.63%	8.28%

Figure 5.6: Profiling of interest-based clusters against session outcome

enough they also are the ones with the highest escalation rate. Another interesting observation is the fact that about 10% of the users who started an escalation did not complete it and ended up either finding a resolution or abandoning the site.

From a Customer Segment standpoint, as Figure 5.7 shows, there's no question that CS_General, being the customer segment with the largest number of sessions (89.8%), is the dominant segment across all intent clusters. It is followed by High Value Seller, which happens to be the second largest customer segment. Perhaps the other interesting call out is that High Value Seller more than doubles its presence within clusters "Topic Browsed focused" (11.44%) and "Escalation focused" (12.77%) compared to its 4.74% global average.

Figure 5.8 shows some interesting differences on the way Desktop and Mobile users are fitted within the intent-based clusters. Notice how most of the intent-

	Popular Solution focused	Topic Browsed focused	Prediction focused	Search focused	Lightly engaged	Popular Solution proficient	Escalation focused	Related Help focused	Super Item Picker focused	Community focused
CS_General	5.78%	6.07%	4.58%	5.93%	50.51%	6.66%	5.05%	5.74%	5.75%	3.92%
CS_Guest	5.71%	2.86%	5.71%	11.43%	40.00%	2.86%	11.43%	11.43%		8.57%
High Value Buyer	6.03%	4.74%	6.47%	6.47%	46.98%	4.74%	10.78%	7.33%	5.17%	1.29%
High Value Seller	3.85%	16.20%	4.98%	6.37%	46.02%	0.40%	15.54%	3.39%	0.93%	2.32%
High Value Trader	3.67%	10.61%	3.67%	7.76%	55.10%	2.45%	10.61%	1.22%	3.67%	1.22%
Managed Account	6.25%	31.25%		6.25%	37.50%		9.38%	9.38%		
New Joiner	8.35%	8.57%	4.03%	3.46%	47.73%	6.55%	7.85%	4.10%	7.34%	2.02%

Figure 5.7: Profiling of interest-based clusters against customer segment

	Popular Solution focused	Topic Browsed focused	Prediction focused	Search focused	Lightly engaged	Popular Solution proficient	Escalation focused	Related Help focused	Super Item Picker focused	Community focused
Desktop	42.32%	75.36%	50.69%	96.84%	65.37%	29.76%	70.88%	96.03%	39.37%	80.41%
Mobile	57.68%	24.64%	49.31%	3.16%	34.63%	70.24%	29.12%	3.97%	60.63%	19.59%

Figure 5.8: Profiling of interest-based clusters against OCS platform

based clusters do not reflect the 65%/35% global split. Instead, some of them like "Search focused" and "Related Help focused" show major differences. The one exception is "Lightly engaged" which shows a split almost identical to the entire population.

	Popular Solution focused	Topic Browsed focused	Prediction focused	Search focused	Lightly engaged	Popular Solution proficient	Escalation focused	Related Help focused	Super Item Picker focused	Community focused
Australia	6.98%	5.56%	3.19%	6.25%	53.02%	7.05%	3.81%	6.87%	7.27%	
Ebay	5.41%	6.01%	4.30%	5.71%	49.47%	7.99%	3.66%	5.18%	6.40%	5.85%
Germany	5.26%	7.60%	4.18%	7.08%	54.78%	4.07%	6.55%	5.87%	3.91%	0.69%
Uk	6.21%	7.84%	6.13%	4.96%	46.05%	4.83%	9.69%	5.13%	4.52%	4.65%

Figure 5.9: Profiling of interest-based clusters against OCS site

Figure 5.9 shows the site distribution by intent-based cluster. For the most part, intent-based clusters show a distribution similar to the global view, with the clear exception of "Popular Solution proficient", "Escalation focused" and Community focused" who clearly show differences on their site allocations.

Chapter 6

Discussion

6.1 Introduction

In this chapter we take a deeper look at some of the results presented in the last chapter in order to derive some key insights. We'd like to call out that the list of insights is by no means exhaustive. Clearly we are interested in showing value by implementing some of most actionable insights, however we are also conscious that the work presented in this practicum is foundational in nature. By that we mean that we have already discussed ways to further the research and also to have business analysts help out with the insights discovery process. For that reason we start this chapter by presenting the main visualization that was used to derive most of the insights presented on this section. As discussed, this visualization was one of the main requirements from the OCS team as they wanted to have a tool that would allow them to explore the new clusters on their own. We then proceed to discuss some of the key findings as well as other interesting call outs.

6.2 Flow diagram and cluster profiler

The flow diagram and cluster profiler is a fully interactive visualization created as a Tableau dashboard whose main purpose is to highlight the main futures

associated with a given cluster, including profiling information. The flow visualization was implemented based on the model created by Olivier Catherin and Jeffrey A. Shaffer (https://community.tableau.com/thread/152115). Figure 6.1 shows a screenshot of the dashboard.

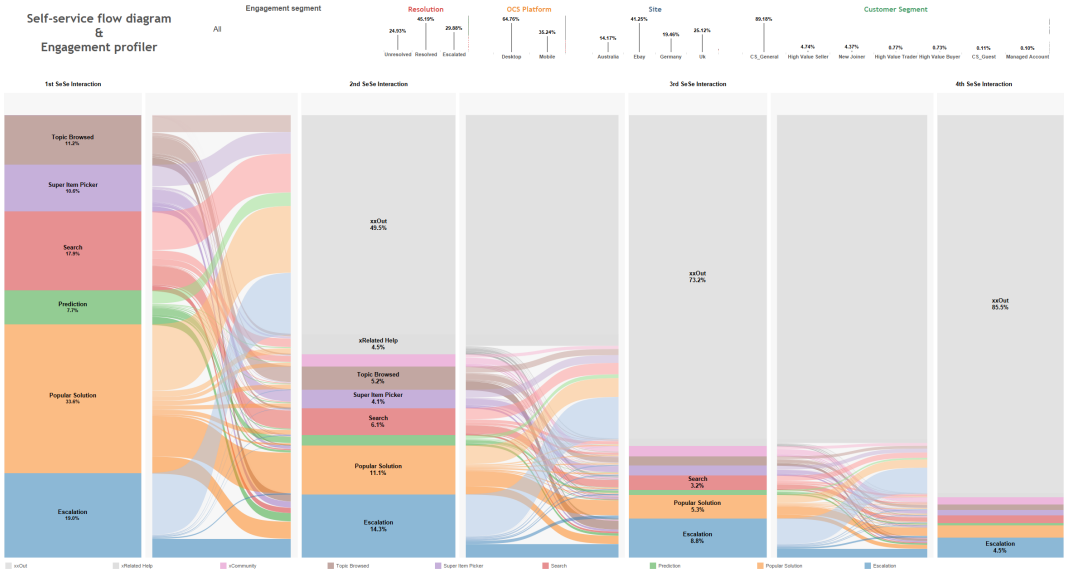

Figure 6.1: Self-service flow diagram and engagement profiler visualization

The dashboard was designed as a visualization that showcases the flow of self-service interactions as they move from segment to segment following the pattern recorded in the sequence-based dataset. The four column charts labeled at the top as, "1st SeSe Interaction", "2nd SeSe Interaction", "3rd SeSe Interaction" and "4th SeSe Interaction" show the percentage of sessions that interacted with a given component within a segment. For example, out of the entire set of sessions, 33.6% interacted with the "Popular Solution" component on their first segment or interaction. Moving on, the second column chart shows that 50.5% of sessions would have had a second interaction. On the other hand, 49.5% of the users would be out of the OCS site. This result is consistent with the segment length analysis where we show that about 50% of all sessions had only one interaction. The dashboard also shows the path that the 50.5% of sessions still engaged would have taken from segment 1 to segment 2. That is, it is meant to show the shift in interaction preference

from segment to segment. Following on the "Popular Solution" example, it can be shown from the Dashboard that about 35% of the original "Popular Solution" population decided to stick with the same component on the second interaction, whereas 65% decided to try a different one.
The second component to the dashboard is the set of for needle charts right at the top. These 4 charts show the distribution of sessions across the profiling dimension. This is meant to provide profiling details for selected interaction-based clusters. To that end, we have included a drop-down selector that allows users to flip between clusters and inspect their properties.

6.3 Key insights

6.3.1 More interactions better resolution?

One key insight that can be derived directly from the Dashboard without even looking at specific clusters is the fact that the ratio of escalated interactions to engaged interactions increases as we move from segment to segment. For example, the escalated to engaged ratio on the first interaction is 19.0%, that is, on the first segment only 19.0% of the population decided to try and escalate their concern. Compare this ratio with the ratio of 28.3% observed on the second segment. Also, it can be shown from the dashboard that the Resolution Rate from people leaving the OCS site after the first interaction is 50.97%. This rate drops to 48.61% for customers leaving the site after their second interaction. Now let's have a look at the dashboard filtered on cluster "Lightly engaged" shown in Figure 6.2.

As expected, 99% of all sessions drop right after the first interaction. The interesting part is that the Resolution Rate for these sessions is 50.88%, which is higher than the global average previously discussed. This means that for the population that have two or more interactions their Resolution Rate in average is slightly lower than those with a single interaction. A result that was truly surprising to the team.

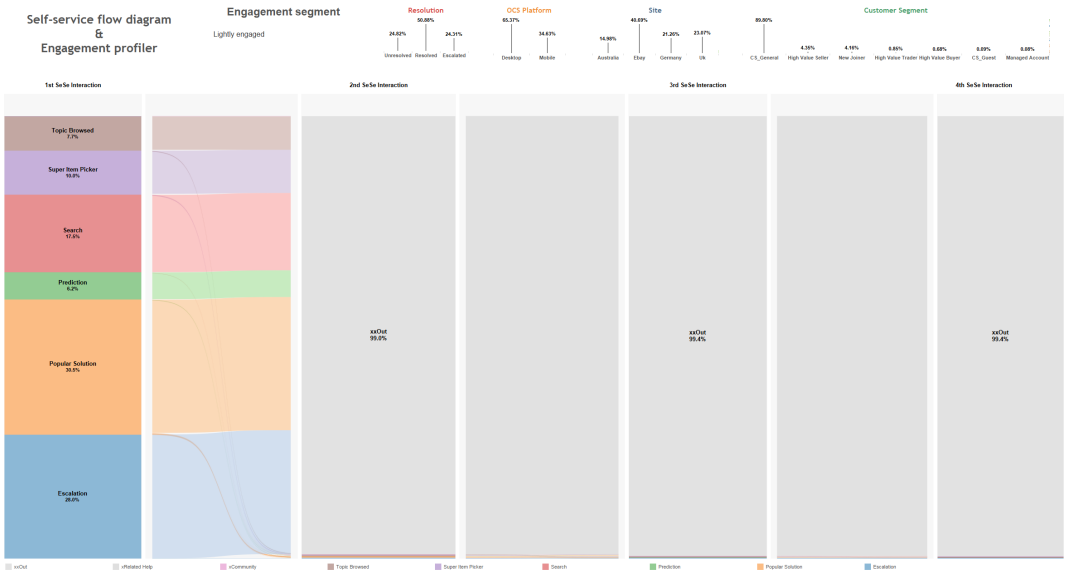

Figure 6.2: "Lightly engaged" interest-based cluster profile

6.3.2 More self-service options better engagement?

Although the OCS team was not as surprised by the groupings found by the clustering algorithm, the result did help reinforce the suspicion that engaged customers mostly interact with the component of their preference rather than trying a different approach. This of course, varies from cluster to cluster as shown by Figure 6.3.

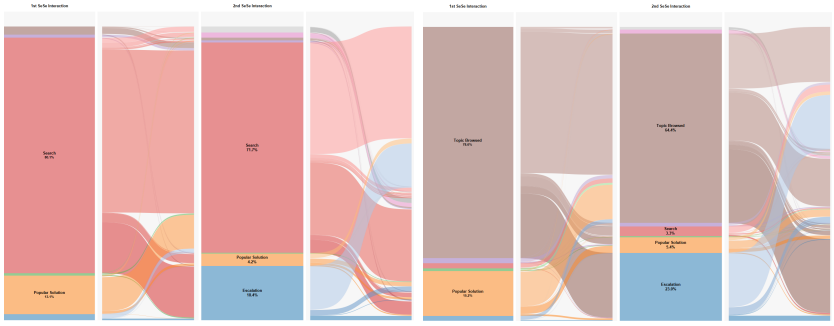

Figure 6.3: "Topic Browsed and Search focused" interest-based cluster profiles

It can be shown from the dashboards in Figure 6.3 that both cluster types show a similar pattern when it comes to the choice of second component for

the second interaction. That is, the repeat interaction rate with the same component. For "Search focused" this rate is 87.91% whereas "Topic Browsed focused" shows 83.69%. This result begs the question as to whether it would be better to try and optimize the experience for a single component instead of providing extra options that most engaged customer seem to ignore is most cases.

6.3.3 Proactively offer escalation options to High Value customers?

We showed in Chapter 5, that the mean session length for customers with multiple interactions was about 3.2 segments per session. We've also shown that customer segment "High Value Sellers" is dis proportionally higher within clusters "Topic Browsed focused" and "Escalation focused". After a few discussions with the OCS team a theory emerged that perhaps "High Value Sellers" showed this particular behavior because they were mostly interested on speaking to a teammate rather than self-serving. This theory still has to be validated, however this bit of info combined with the low mean session length and the observable drop in Resolution Rate in later segment interactions, made us wander whether the OCS team should try and optimize the experience for High Value Sellers and proactively offer an escalation channel after having 5 or more self-service interactions.

Chapter 7

Conclusions and Future Research

7.1 Summary

As stated in the Introduction, the aim of this practicum was to provide the OCS team at eBay with detail information on the components that make up the new OCS Engagement metric and a tool for them to conduct their own analysis and derive their own insights. We accomplished this by providing an in-depth analysis of what self-service interactions look like and how they differ among them, specifically providing data on main statistics around OCS session length and duration, which showed some surprising results (higher than expected session length and lower than expected session duration). We were also able to demonstrate that there are cluster patterns in the self-service interactions data and that they group around 10 clusters. We also characterized these clusters and provided insights in regards perceived performance from a customer experience standpoint, showing that having multiple interactions does not result in clear Resolution Rate increases. We also showed that engaged customers, those with two or more interactions are more likely to use the same self-service option rather than switching to another solution. Finally, we were able to show that all the information and new insights might be gathered

from the Tableau Dashboard that was designed for this project.

7.2 Contributions

The main contributions to practice from this work can be summarized in two buckets. The first one is supported by the insights obtained with this work that point out opportunities to improve the overall customer experience. Opportunities range from fast-tracking access to live channels to High Value customers that show signs of friction after surpassing a pre-defined session length threshold defined by the business team. Opportunities to influence design decisions that align with the observable desire of customers to engage with the same self-service component rather than use a different one. Another opportunity relates to optimizing OCS Platforms (desktop vs mobile) based on the clear preferences expressed by the different cluster profiles. Also an opportunity to proactively reach out to High Value customers that we know engage only once and did not get a resolution to their problem. The other contribution aspect from this work is a more pragmatic one in the form of a Dashboard and visualizations that open up possibilities for the team to dig even further to extract more insights that otherwise would be much harder to accomplish with the limited time and resources dedicated to this practicum.

7.3 Future work

The first natural extension to this work would be to scale up the application to include other regions and more data into the analysis. The second extension would be to include site content information into the analysis. Up until now, the analysis has exclusively focused on the mechanics of the self-service interactions. Including information on the actual content presented to customers would help to come up with more refined segments and other type of insights related to content preferences. Finally, move on to complete the profiling of common paths using the sequence-based representation clustering results which could not be completed on time for this Practicum.

Appendix – Source Code

All source code used for this Practicum including, SQL code, R scripts, Tableau source file and Latex source files are available at the following URL:

https://github.com/edgarcadena/Practicum

Download file **Source Code.zip** to get access to the source files.

Appendix – Interest-based profiles

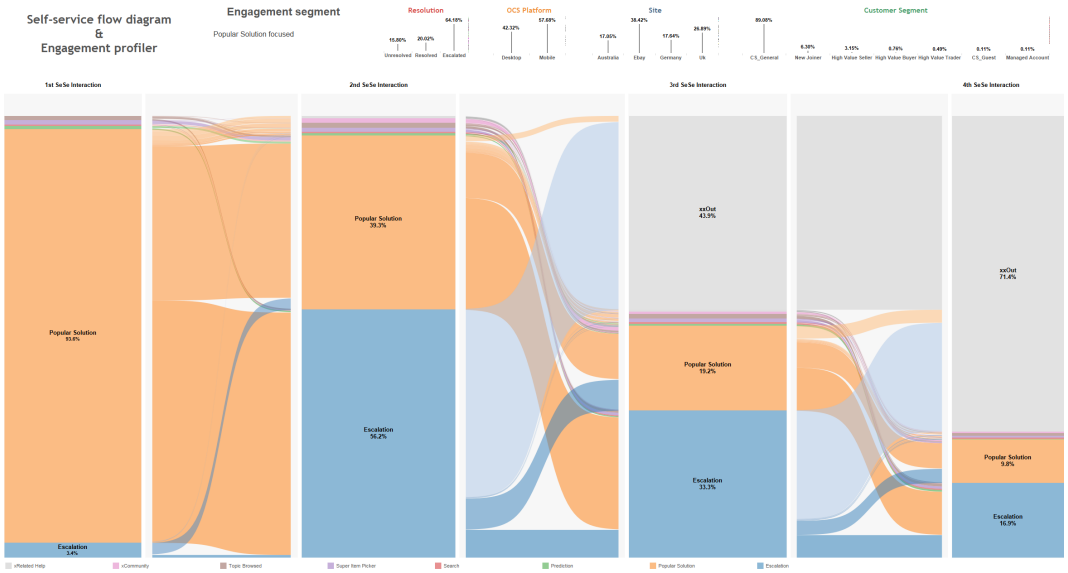

Figure 7.1: "Popular Solution focused" interest-based cluster profile

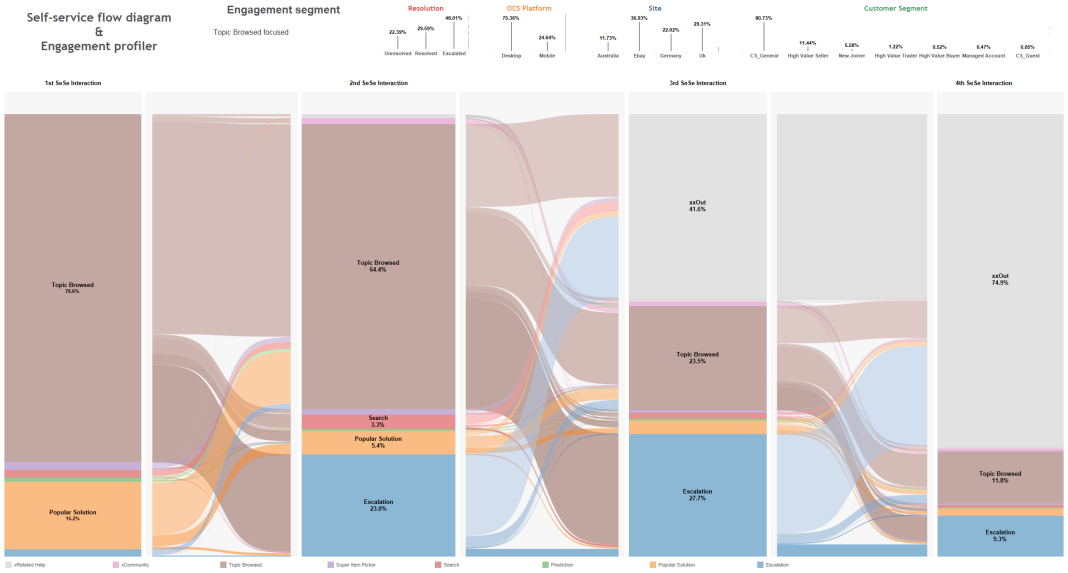

Figure 7.2: "Topic Browsed focused" interest-based cluster profile

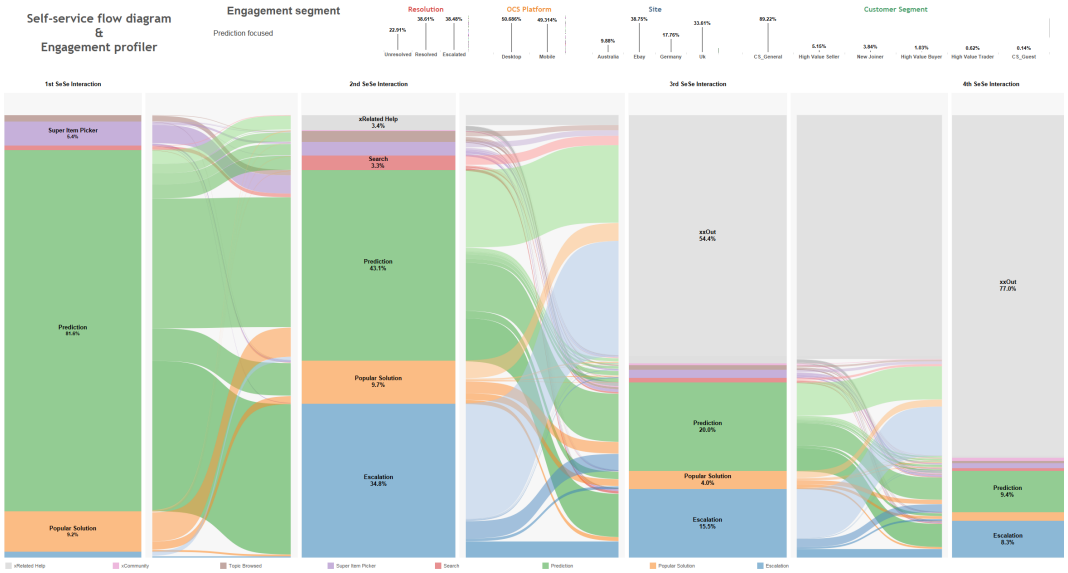

Figure 7.3: "Prediction focused" interest-based cluster profile

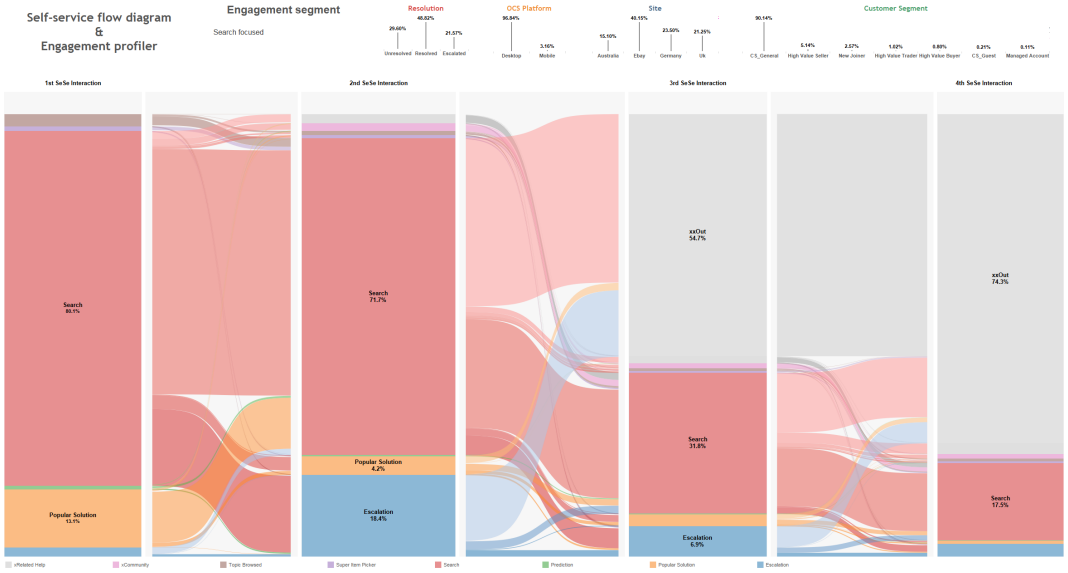

Figure 7.4: "Search focused" interest-based cluster profile

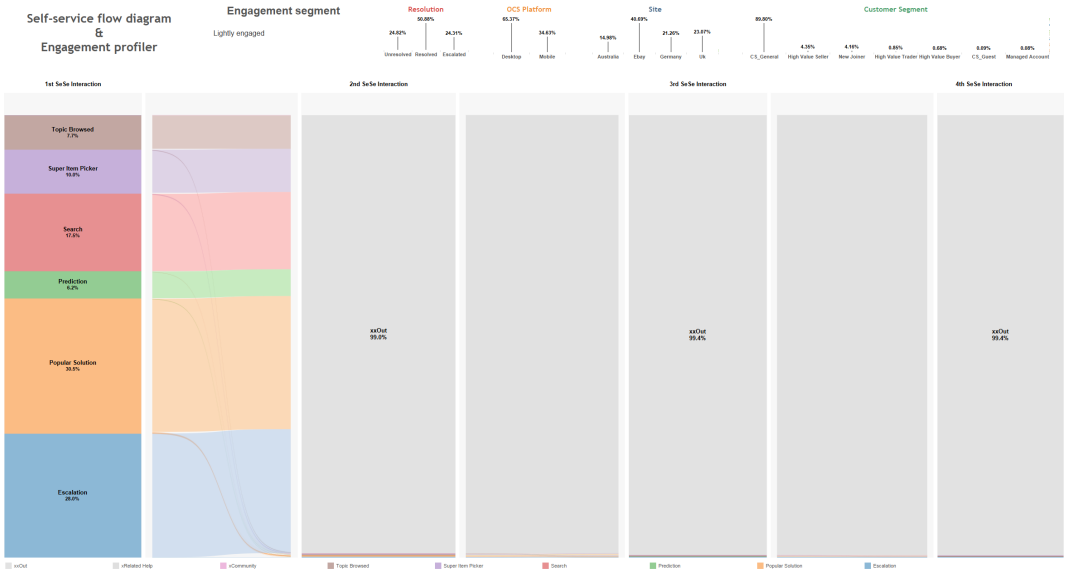

Figure 7.5: "Lightly engaged" interest-based cluster profile

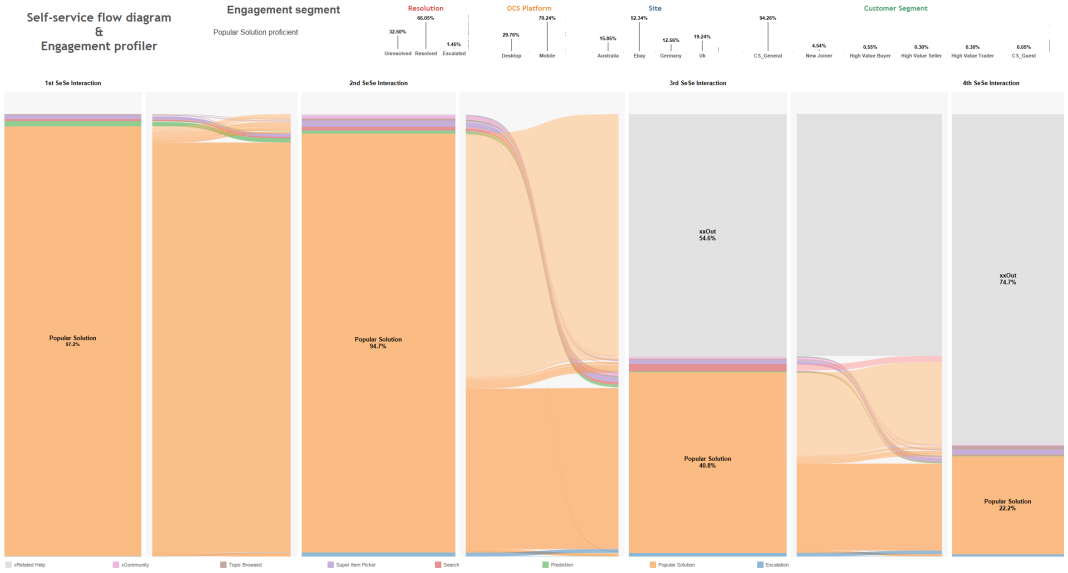

Figure 7.6: "Popular Solution proficient" interest-based cluster profile

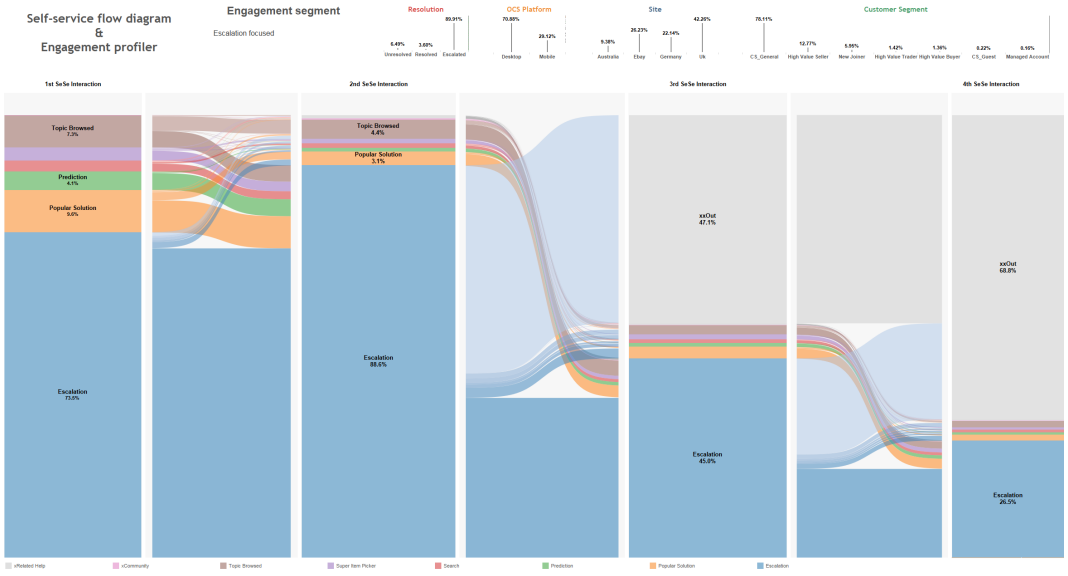

Figure 7.7: "Escalation focused" interest-based cluster profile

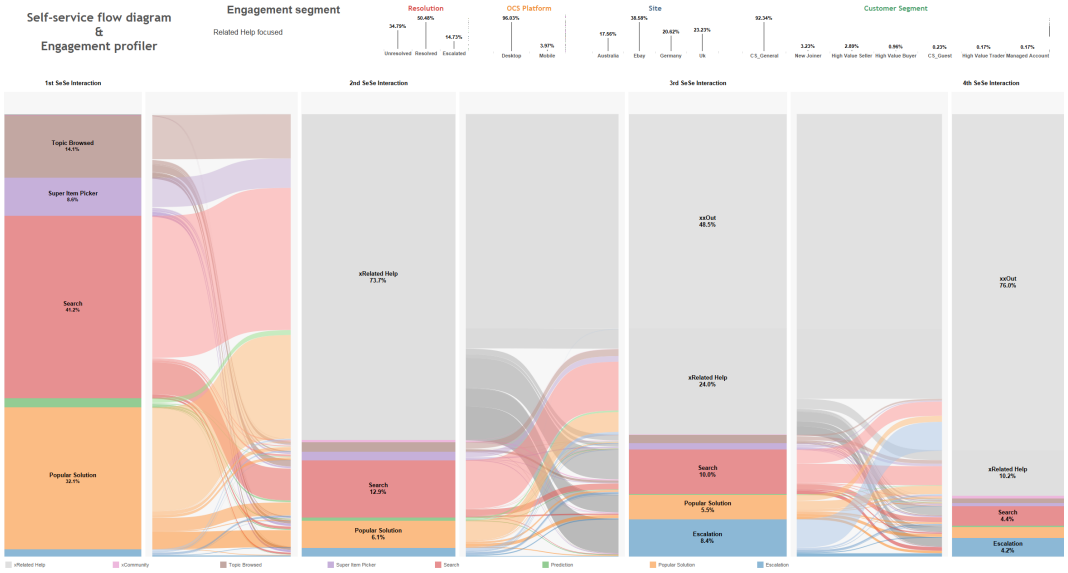

Figure 7.8: "Related Help focused" interest-based cluster profile

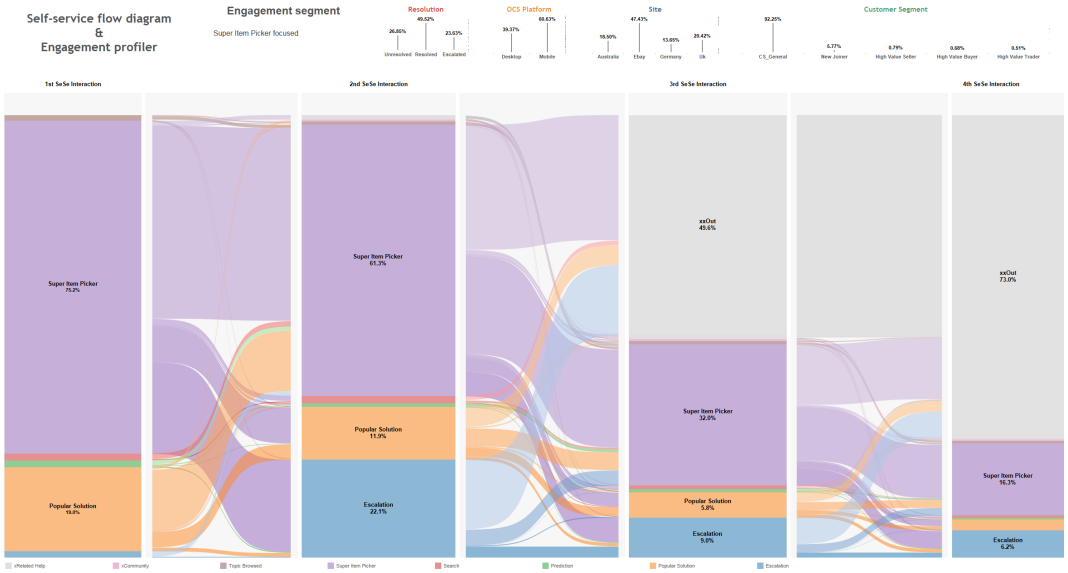

Figure 7.9: "Super Item Picker focused" interest-based cluster profile

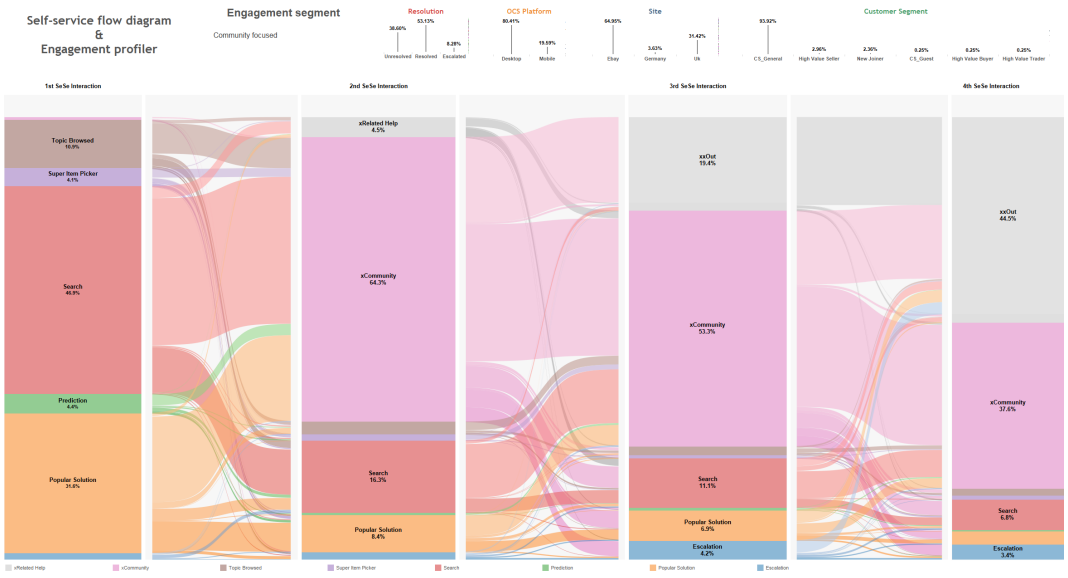

Figure 7.10: "Community focused" interest-based cluster profile

Bibliography

Albuquerque, P., S. Alfinito and C. Torres. 2015. Support vector clustering for customer segmentation on mobile tv service. *Communications in Statistics-Simulation and Computation*, **44**(6): 1453–64.

Davenport, T. and P. Klahr. 1998. Managing customer support knowledge. *California Management Review*, **40**(3): 195–208.

Dhandayudam, P. and I. Krishnamurthi. 2014. A rough set approach for customer segmentation. *Data Science Journal*, **13**: 1–1.

Gusfield, D. 1997. *Algorithms on strings, trees and sequences: computer science and computational biology.*

Hamka, F., H. Bouwman, M. de Reuver and M. Kroesen. 2014. Mobile customer segmentation based on smartphone measurement. *Telematics and Informatics*, **31**(2): 220–7.

Hung, Y., K. B. Chen, C. Yang and G. Deng. 2013. Web usage mining for analysing elder selfcare behavior patterns. *Expert Systems with Applications*, **40**(2): 775–783.

Kaufman, L. and P. J. Rousseeuw. 2009. *Finding groups in data: an introduction to cluster analysis*, volume 344.

Liu, H. and V. Keselj. 2007. Combined mining of web server logs and web contents for classifying user navigation patterns and predicting users' future requests. *Data and Knowledge Engineering*, **61**(2): 304–30.

Negash, S., T. Ryan and M. Igbaria. 2003. Quality and effectiveness in web-based customer support systems. information and management. *California Management Review*, **40**(8): 757–68.

Olatz, A., I. Gurrutxaga, A. Lojo, J. Muguerza, J. M. Perez and I. Perona. 2013. Web usage and content mining to extract knowledge for modelling the users of the bidasoa turismo website and to adapt it. *Expert Systems with Applications*, **40**(18): 7478–91.

Pang-Ning, T., M. Steinbach, V. Kumar *et al.*. 2006. *Introduction to data mining*, volume 74.

Smith, W. 1956. Product differentiation and market segmentation as alternative marketing strategies. *Journal of Marketing*, **21**: 3–8.

Truel, O. and C. Connelly. 2013. Too busy to help: antecedents and outcomes of interactional justice in web-based service encounters. *International Journal of Information Management*, **33**: 674–683.

Yao, Z., P. Sarlin, T. Eklund and B. Back. 2014. Combining visual customer segmentation and response modeling. *Neural Computing and Applications*, **25**(1): 123–134.