Generation of Personalized E-mail Subject Lines

Msc Thesis Computer Science & Engineering

Stef Kaptein

TU Delft

Generation of Personalized E-mail Subject Lines

Msc Thesis Computer Science & Engineering

Thesis report

by

Stef Kaptein

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on May 28, 2024

Thesis committee:	Pradeep Murukannaiah
Chair:	Sole Pera
Supervisor:	Sole Pera
Place:	Faculty of Electrical Engineering, Mathematics, Computer Science, Delft
Project Duration:	From September 4, 2023 - To May 28, 2024
Student number:	4683919

Faculty of EEMCS · Delft University of Technology

Delft
University of
Technology

Abstract

The online presence of e-commerce platforms, publishers, and other businesses has been increasing with the rise of internet use. These companies regularly expose their (potential) customers to massive amounts of media and products. It is vital for these companies to elicit interactions from these individuals in order to sell products or subscriptions. Engagement with online items, such as articles or products, are often measured using clicks. A well-known and proven method for improving online interactions is the use of clickbait. However, this practice is known to degrade the reputation of businesses. Widespread use of obvious clickbait is, therefore, not a sustainable solution for reputable companies. Research into alternative solutions to foster engagement is mostly taken from the perspective of the consumer. These solutions mainly involve various recommender algorithms that match consumers with items and articles that interest them. However, research from the perspective of the producer is extremely limited. The producer of the content has different incentives than the consumer of the content. Their main goal is to elicit an interaction from the consumer, while the consumers' main goal is to find something that interests them.

In this thesis, inspired by the work on personalization for consumers, we make the producer the main stakeholder. We explore whether the personalization of subject lines for groups of users with similar behaviour patterns and interests improves the engagement of consumers on newsletters. With the digital marketing company Basedriver as the use case, we personalized the headlines of marketing e-mails in newsletters and performed experiments to analyse the effectiveness of our technique. We hypothesise that e-mail marketeers can improve engagement by personalizing subject lines to address the personal interests of consumers. By looking at the content of articles previously clicked by consumers, they can be grouped into user groups with similar interests. Information on the interests of users from these groups can then be leveraged to generate a subject line addressing the interests of the group. The historic customer behaviour data and platform used for the validation of our new personalization strategy was provided by Basedriver and Hearst, who assisted in the research. Basedriver also provides the platform for collecting the data from the experiment.

Preface

Contents

List of Figures

List of Tables

1
Introduction

Online media consumers get exposed to more content than they can consume daily. A substantial portion of this content is marketing material, for example, advertisements and sponsored messages. The landscape of online marketing is dynamic, characterised by continuous innovation and the emergence of new, fresh ideas. Despite being considered old-fashioned by some, email marketing remains a powerful tool for reaching your target audience and creating engagement with your products or content. Just as in other forms of online media such as online news [1, 2, 3], a critical component of the success of email marketing campaigns is the effectiveness of the subject line. The subject line is the first point of contact and influences the recipient's decision to open the email [4, 5]. In addition, the subject line sets the expectations for the reader, which, if not met, can leave a reader annoyed and hurt the readers view of the company [6].

Marketers use various strategies to make people interested in their emails. Tricks like emojis or personalised greetings can make the subject line more appealing. For example, adding a recipient's first name can make them more likely to click. However, such strategies become less effective as people become accustomed to them. Another technique that marketers might use is called clickbaiting. Click baiting can encompass many techniques, aiming to lure readers in with sensational or exaggerated claims or by making promises the content cannot live up to. While it might initially attract clicks, clickbait can harm a brand's reputation in the long run [7]. Readers may feel misled or disappointed when the content doesn't live up to the promises made in the subject line, leading to a loss of trust and credibility. In email marketing, building trust is crucial for establishing lasting relationships with the audience, making clickbait counterproductive.

Writing a subject line that grabs the attention of as many readers as possible is not trivial, and marketers put a lot of effort and thought into it. Marketers mostly work based on proven tactics and experience. For example, they might look at the subject lines of successful previous mailings. Marketers perform A/B tests on a random subset of their recipients to compare different subject lines and find the best one. They send them to small groups of users and measure the open and click rates. Although this might indicate how well a subject line resonates with (a part of) the audience, there is no guarantee that this group represents the entire audience. The target audience is diverse, with different behaviour patterns and interests. It is impossible to effectively address the entire target audience using the same subject line. Segmenting the audience into groups with similar behaviours and interests allows for better targeting. However, this is infeasible to do without the proper tools and expert knowledge.

This presents a challenge to the marketer: How can the target audience be segmented so that they can be more effectively targeted? This problem is not unique to email marketing. All forms of digital marketing are continuously improving how they reach their target audience as effectively as possible. Following the ever-increasing use of artificial intelligence (AI) in the past few years, there has been a rise in the use of machine learning (ML) in digital marketing for analysing huge data sets and using this for predictions, recommendations, personalisation, and more [8]. With this increased use, much research is being done on ML applications in digital marketing [9, 10, 11]. However, literature on ML applications in email marketing remains scarce, and the existing literature is based on small datasets, which limits the applicable ML techniques [12].

Studies mostly focus on predictive analysis, such as predicting open and click rates [13] or predicting campaign success [14]. The analysis is also mostly based on individuals, which is not scalable and privacy-sensitive. For example, in [15], recipients are profiled based on identifiable information such as location and device. Existing research has shown the comparisons between several ML techniques on a bigger dataset to predict email open rates, demonstrating how much information can be extracted from larger datasets [12]. These works provide tools and ideas for marketers to predict the performance of their campaigns, content and subject lines. However, marketers must still create the content and subject line themselves. There is little existing research on the generation of subject lines. Zhang et al. introduced the email subject line generation task in 2019 [16], but there has been little published work on this task since. In addition, no information on the recipient is considered when generating the headline. This illustrates a clear gap, where the information gathered from the user's previous behaviour is leveraged to generate personalised subject lines.

Many publishers send out recurring newsletters to their readers, which contain articles with varying topics. Readers of these newsletters are not all interested in the same topics. When writing a subject line, the marketer picks the article they think will interest the most people, consequently lowering the change of engagement for the group of users not interested in this topic. We argue that if publishers were to maximise the engagement of each user on their newsletter, they would ultimately be able to reach people with their content and products while avoiding practices like clickbait. With that in mind, in this work, we focus on taking a step towards maximizing user engagement, starting by modelling and grouping users with similar behaviours and interests. By leveraging the interests of the groups, we can personalise email subjects based on content selected using the group interests. By personalising the subject line to fit the interests of each group, we predict that the overall engagement of the group will increase. For this, we work together with the marketers of Quest, a Hearst Publishers brand. They sent their weekly newsletter to 93.000 people and used this exact strategy to create the subject line for their weekly newsletter

To address this problem, we propose to apply ML techniques to create user groups from traits and interests extracted from historical clicking behaviour and generate personalised subject lines for these groups. We defined the following research objective:

Research Objective

Identify user groups based on historical clicking behaviour with emails and leverage this information to generate personalised headlines that increase user engagement.

To guide the research and ultimately attain the research goal, we formulate 3 research questions:

Research Question 1

How can we model user behaviour patterns and interests based on past interaction with marketing emails?

We analyse the user's clicking behaviour and interests and create an individual user model based on features extracted from historical clicking data. Based on preliminary analysis, we identify several features based on which we can model an individual user. The features represent user behaviour, such as click and open counts and high-level interests based on the article categories defined by the writer.

Research Question 2

How can users be grouped based on behaviour patterns and interests?

Based on the features identified by the user analysis, we examine how to group users with similar behaviour and interest patterns. For this, we create clusters, which we can leverage to create personalized subject lines that cater to the user group's interests and increase engagement. Thus, we explore which clustering strategy best suits our needs and features.

Research Question 3

Do subject lines personalised based on user group interests increase engagement?

To address this RQ, we first leverage similar interests and behaviors observed on aforementioned clusters to generate a subject line for the article that will most likely prompt engagement from the users in the group. For this, we devise a strategy for selecting an article from the newsletter most likely to prompt engagement and a strategy for generating a subject for this article.

To validate whether personalised headlines increase engagement, we conducted an experiment in collaboration with 2 partners: Quest, a brand belonging to Hearst Publishers and Basedriver, an email marketing platform, which will facilitate the sending of the emails and provide us with the data. The experiment aims to explore the impact of personalising for user groups on engagement.

The key contributions of this study are outlined as follows.

- Improved user segmentation based on diverse features extracted from clicking behaviour only.
- Personalised subject line generation using state-of-the-art Large Language Model (LLM) technology based on the user group's interests.
- 'In-the-wild' experiment on real users to analyze the effect of personalised subject lines on clicking behaviour.

Before describing the methods used in our exploration, we first give the required background information on the company and discuss some related work. We then discuss the methods used to guide our exploration and discuss our experiment. Finally, we wrap up with a discussion on the results of the experiment and the limitations of our work.

2

Background and Related Work

In this Chapter, we provide the necessary background information on the company we are working with to create and send the emails and discuss related work. For this research, we are working with two companies: Hearst and Basedriver. Hearst is a publisher which maintains several brands which publish magazines, sell products and send newsletters. Basedriver is the software they use to send emails and collect the results. For this purpose, Basedriver maintains profiles of people receiving Hearst's emails. Hearst uses Basedriver to create the emails and select and send the emails to the correct profiles. To understand where the historical clicking data comes from and how emails are constructed and organised, we discuss how these elements are collected and created within the context of the Basedriver software. We also discuss work related to digital marketing and email subjects.

2.1. Company Background

This research was performed in collaboration with Basedriver[1]. To understand the background of the data and the specific implementation of the research, it is crucial first to understand the role that Basedriver fulfils in email marketing. Basedriver sells its software as SAAS to publishers in the Netherlands and Belgium. These publishers often have multiple brands with separate marketing teams that want to send newsletters or other mailings to their customers. The power of Basedriver is its ability to personalize mass mailings based on customer profiles. For example, adding a certain content section for a customer with a certain subscription, while someone else will not get this content. All of this happens in the same email campaigning. Basedriver can also be adjusted to a publisher's needs. For example, the type of content and the make-up of this content can be adjusted as per the publisher's requests. This makes Basedriver a powerful tool for marketers who want to reach a broad range of customers.

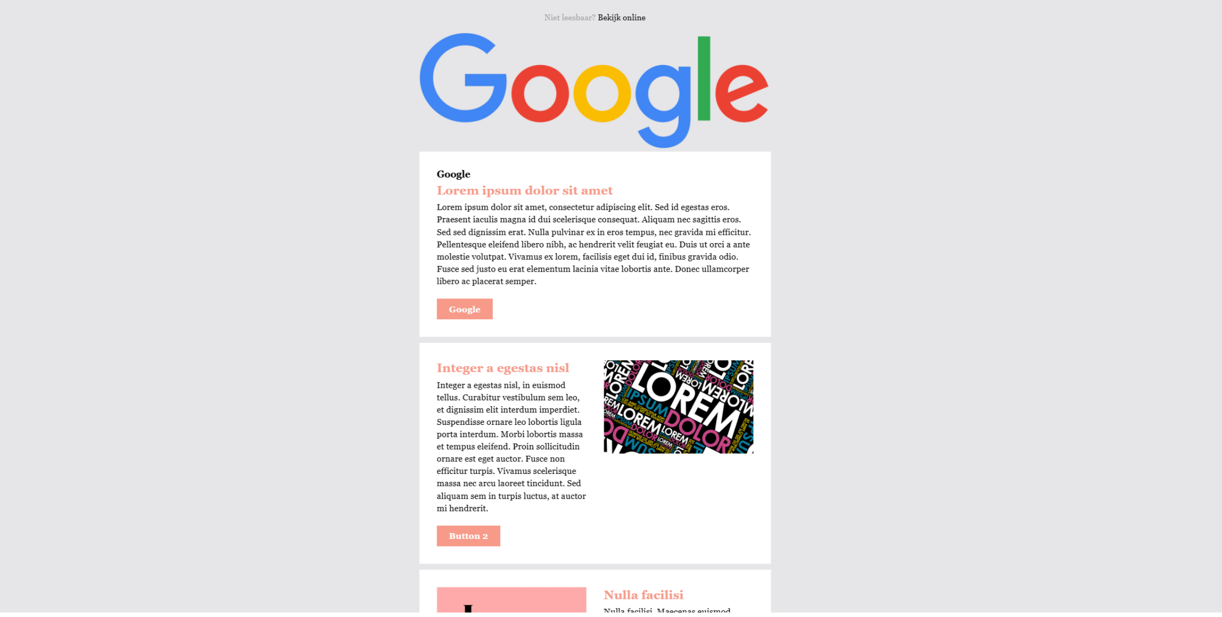
Basedriver also collects all the statistics of the sent mailings, for example, clicks and opens. These statistics are all linked to the profiles in the database. Basedriver can use this to see if a user has already received a piece of content. But it can also be used by marketers to send follow-ups to customers who have already received a particular piece of content.

To give the reader a better idea of what the software exactly does, how it works and how a personalization strategy would be implemented, I will give a rundown of the most important components that make up an email campaign. Basedriver has many aspects that I will not discuss or explain or only briefly touch on; this is done to avoid giving too much unnecessary information for the context of this research. All images and examples that are shown are from the same publisher that was used for the experiment.

2.1.1. Email

The goal of Basedriver is to eventually send an email to the customer with which the customer interacts. An email in Basedriver is built up from several content items (these will be discussed in more detail in the next section). Figure 2.1 shows an example of such an email. Basedriver's internal systems decide which content articles are selected for every customer. In practice, you get several groups with similar characteristics that get different emails. The email contains several buttons and images that the customer can click to lead to the brand's website. They could, for example, lead to the full article that the newsletter article is referencing. How these clicks work in detail is discussed later in this section.

[1] https://www.basedriver.com/en/



Google

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id egestas eros. Praesent iaculis magna id dui scelerisque consequat. Aliquam nec sagittis eros. Sed sed dignissim erat. Nulla pulvinar ex in eros tempus, nec gravida mi efficitur. Pellentesque eleifend libero nibh, ac hendrerit velit feugiat eu. Duis ut orci a ante molestie volutpat. Vivamus ex lorem, facilisis eget dui id, finibus gravida odio. Fusce sed justo eu erat elementum lacinia vitae lobortis ante. Donec ullamcorper libero ac placerat semper.

Integer a egestas nisl

Integer a egestas nisl, in euismod tellus. Curabitur vestibulum sem leo, et dignissim elit interdum imperdiet. Suspendisse ornare leo lobortis ligula porta interdum. Morbi lobortis massa et tempus eleifend. Proin sollicitudin ornare est eget auctor. Fusce non efficitur turpis. Vivamus scelerisque massa nec arcu laoreet tincidunt. Sed aliquam sem in turpis luctus, at auctor mi hendrerit.

Nulla facilisi

Nulla facilisi. Maecenas euismod

Figure 2.1: Example of a newsletter email. You can see the header image (the Google logo) and several articles in the newsletter. Images and buttons can be clicked.

2.1.2. Content

As explained in the previous section, an email is built up from content articles. Marketers working for the publishers create these content articles in Basedriver. This interface can be seen in figure 2.2. This particular example is for the 'nieuwsbrief header' content type, which translates to 'newsletter header'. There are several content types; the main difference between them is the fields that they have and how they are rendered. You could have another content type without an image, for example, or one without a button. As can be derived from the name, the content type in the figure is also the header article. This is a special type of content as it defines the title and snippet of the email that will be sent.

2.1.3. Profiles

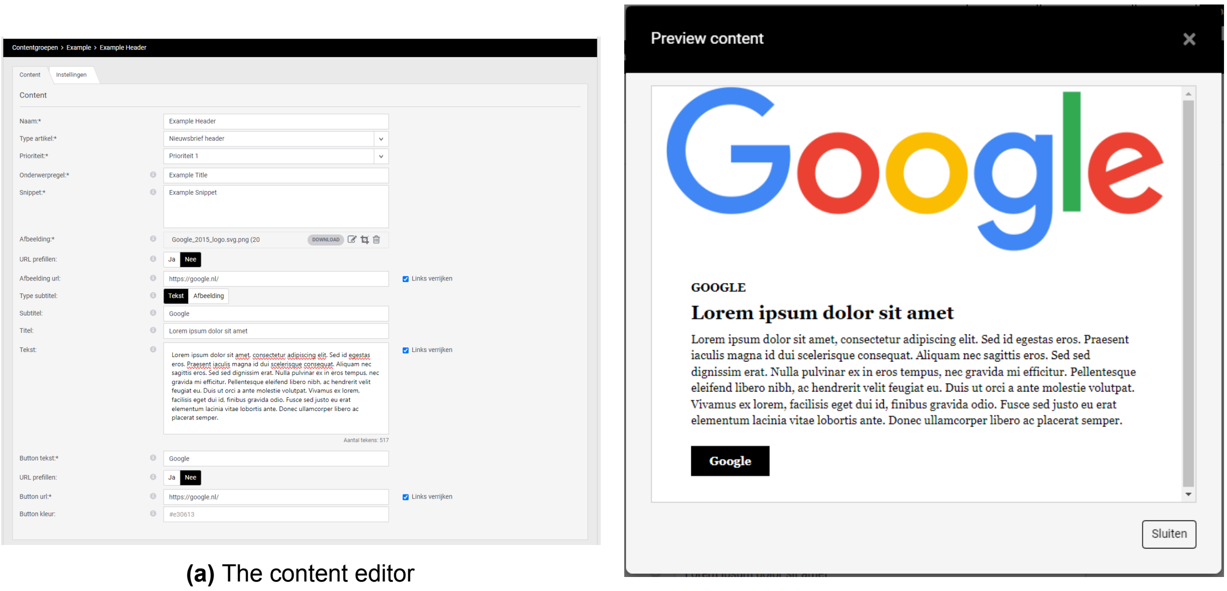
Basedriver has a database of all the customers of a publisher. This database is updated every day with data that is supplied by the publisher. In addition, Basedriver has an API which allows publishers to set up forms for opting into newsletters or other forms of email marketing. A profile in Basedriver may have opt-ins or subscriptions for multiple brands from the same publisher. Therefore, click information collected from one brand of the same publisher may be used for better-targeted mailings of another brand. This greatly increases the amount of data that is available to personalize. A profile record in the database also contains all the previous mailings and the actions performed on those mailings by the customer.

2.1.4. Email contact results

After a mailing has been sent, Basedriver starts collecting the results from the customers. There are four possible results in Basedriver: sent, opened, bounced and clicked. Sent and opened speak for themselves. However, the bounced result might require some further explanation. A bounced result means that the customer did not receive the email for whatever reason. This could be due to a full mailbox, a user marking a mail as spam or an unknown email address. The send, opened and bounced results are mutually exclusive for a single mailing. A mail can have multiple clicks, however. When a mail contains multiple links, these all produce unique results; the same is true for multiple clicks on the link. These results are stored in the database and exported daily in incremental files.

2.2. Related Work

As discussed in the introduction in chapter 1, there is, to our best knowledge, very little literature on the generation of personalised email headlines. However, this does not mean that no literature is related to our



Contentgroepen > Example > Example Header

Content | Instellingen

Content

Naam:* Example Header

Type artikel:* Nieuwsbrief header

Prioriteit:* Prioriteit 1

Onderwerpregel:* Example Title

Snippet:* Example Snippet

Afbeelding:* Google_2015_logo.svg.png (20 DOWNLOAD

URL prefillen: Ja [✓ Nee]

Afbeelding url: https://google.nl/ ☑ Links verrijken

Type subtitel: [✓ Tekst] Afbeelding

Subtitel: Google

Titel: Lorem ipsum dolor sit amet

Tekst: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id egestas eros. Praesent iaculis magna id dui scelerisque consequat. Aliquam nec sagittis eros. Sed sed dignissim erat. Nulla pulvinar ex in eros tempus, nec gravida mi efficitur. Pellentesque eleifend libero nibh, ac hendrerit velit feugiat eu. Duis ut orci a ante molestie volutpat. Vivamus ex lorem, facilisis eget dui id, finibus gravida odio. Fusce sed justo eu erat elementum lacinia vitae lobortis ante. Donec ullamcorper libero ac placerat semper. ☑ Links verrijken

Aantal tekens: 517

Button tekst:* Google

URL prefillen: Ja [✓ Nee]

Button url:* https://google.nl/ ☑ Links verrijken

Button kleur: #e30613

(a) The content editor

Preview content ×

GOOGLE

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed id egestas eros. Praesent iaculis magna id dui scelerisque consequat. Aliquam nec sagittis eros. Sed sed dignissim erat. Nulla pulvinar ex in eros tempus, nec gravida mi efficitur. Pellentesque eleifend libero nibh, ac hendrerit velit feugiat eu. Duis ut orci a ante molestie volutpat. Vivamus ex lorem, facilisis eget dui id, finibus gravida odio. Fusce sed justo eu erat elementum lacinia vitae lobortis ante. Donec ullamcorper libero ac placerat semper.

(b) Content preview

Figure 2.2: What a marketer uses to create and preview the content. This is the same content that is displayed in the top content article in the example email in Figure 2.1

research or trying to achieve a similar goal. In this section, we discuss several related works; it is divided into two main parts: first, digital marketing. Digital marketing is the encompassing field of email marketing and includes online advertising or stores. Since these fields are closely related, recent works can provide some comparison to our work. Second, email subject. The literature on email subjects, especially predicting performance, is extensive. However, literature on generation, specifically personalised generation, is virtually non-existent.

2.2.1. Digital marketing

One of the most difficult parts of a marketer's job is to reach the audience most interested in the product, subscription, or content they are selling. Before the internet age, all marketing was done 'offline', such as advertisements in public spaces, newspapers, or magazines. However, because these kinds of advertisements reach a large group of diverse people, they can be expensive and ineffective. In addition to your target audience, you also reach a large group uninterested in your product. With the rise of the internet in the past 30 years came a new form of marketing: digital marketing. Digital marketing allows marketers to reach their target audience everywhere at any time. More recently, we have seen the rise of big data and Artificial Intelligence (AI). Marketers in all forms of marketing have found ways to use AI to increase their productivity and efficiency [8]; for example, using AI algorithms to analyse a user base and segment it based on behaviour [17, 18], prediction of user behaviour [19, 20] or personalisation of user experience [21].

2.2.2. Email subjects

One form of digital marketing is email marketing. Although it is a very different medium than other forms of marketing, it poses similar challenges: predicting campaign success and creating content to reach your target audience effectively. Email marketing is relatively cheap compared to other forms of digital marketing, leading to its widespread use. It is no coincidence that the average open and click rates of email marketing campaigns are low. As explained in the introduction, the subject line is critical to the success of an email campaign. Therefore, most research on email marketing focuses on the subject line.

Subject performance prediction

For example, predicting the open rate of a subject line based on company (the company running the email campaign) and subject line textual features [12] or using email and recipient features [15, 14]. These works also extensively use data mining techniques such as a bag of words or TF-IDF to extract and count keywords used in successful past campaigns [13]. Although these works provide insights into the different factors that determine a subject line's success and can assist marketers in evaluating their subject lines, they still require marketers to write the subject lines themselves.

3 Methods

In this Chapter we discuss the methods used to answer our research questions. We discuss the different steps, from the analysis of user clicking behaviour to the experiment, in chronological order. Before we can perform the analysis of the historical clicking behaviour, we first need to collect the clicks and pre-process them such that they can be analysed, this is discussed in Section 3.1. After pre-processing the clicks, we can analyse the behaviour to determine patterns and select features, from which we can extract interests and behaviour features to segment the users into groups. The analysis and feature selection are discussed in Sections 3.2 and 3.3, respectively. Using the extracted features, we perform clustering to segment the users into user groups with similar interests and behaviours, for which we can personalise the subject line to increase engagement, we discuss the user grouping in Section 3.4. Using the interests and behaviour features, we group users into groups with similar interests. To increase engagement, we leverage the information of the group's interests into creating a subject line highlighting these interests from the candidate email; we discuss this personalization strategy in Section 3.5. Firstly, we discuss the pre-processing of the source data.

3.1. Data pre-processing

In this section, we discuss the source of historical click data for the profiles of Hearst. The historical click data contains information on the behaviour of the users and the articles they have engaged with, which we can later use to extract interest. We also look at the pre-processing done on the data to make it ready for analysis and further processing in the user model. First, we present an explanation of the source of the data, how it was retrieved and the period over which the data was collected. Then, we discuss how the data was pre-processed and why this was necessary.

3.1.1. The source data

The data required for this research can be divided into the contact results and the content. The files and their respective sizes are in table 3.1a. The contact results contain all the clicks and opens, together with the date and the profile to which it belongs. This data was all provided by Basedriver and goes back to July 2019. Basedriver also provided the content; however, this only included the shortened segments of larger articles in the email newsletter. The full articles were scraped from the brands' websites to gather information on the topics (or categories) that the content belongs to. This resulted in a file with article URLs and their respective categories and the HTML content of the website. This could then be further processed to get the cleaned content of the articles. Because a click result contains the URL clicked on, these two files can be joined together on the URL. Thus, getting all the clicked content for a single profile for use in the user behaviour and interest analysis is possible.

3.1.2. Contact Results

As explained in Section 2.1.4, contact results are stored in the database under the profiles. Thus, it might seem like retrieving the data is as trivial as exporting the database as-is. However, this is not the case. Over the years this client has been using Basedriver, the number of campaigns and mailings they have sent have stacked up. Not all information in these results is relevant after a few months. Therefore, Basedriver performs a cleanup every few months, archiving all old campaigns and the related contact results. Because Basedriver also provides consultancy and reporting services to its clients, exports that contain the data

Source file	Size	Number of rows
all_contact_results.csv	23.778 GB	359,972,302
cleaned_content.csv	77 MB	22,282

(a) The source files after pre-processing

BaseDriverId	The internal ID used within Basedriver to identify emails
Done	The date the mailing was sent
ProfileId	The ID of the profile that this action belongs to
Action	The action; Sent, Open or Click
ActionDate	The date of this action
Key	If the action was a click, this is where the URL is, otherwise it is empty

(b) Columns in the contact results file

Figure 3.1: Contact Results and Data Files

needed for this research are available. However, these were only available for the most recent months. These export files needed to be supplemented with the archived campaign data to complete the data.

The exported files were in a format that could be easily used to link the clicks of the profiles with the content. The format can be seen in Table 3.1b. The archived campaigns were in JSON format and contained much redundant information. These JSON files were parsed and transformed to the same format as the export files. That way, they could be easily joined to create one big CSV file containing all the results for the past 5 years. This gives us the most information possible, since engagement on emails is generally low, we want to maximise the number of interactions we have available for analysis and modelling.

3.1.3. Content

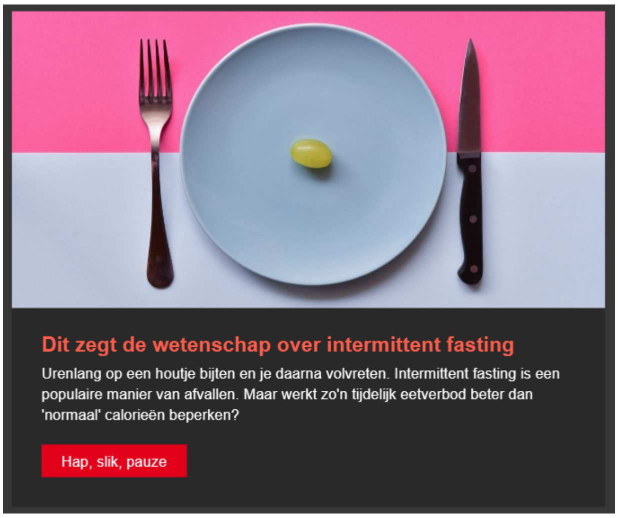
A newsletter email contains short snippets of multiple articles. Marketers do not put full articles in their emails for two reasons: firstly, it becomes too long and cluttered if you put five full articles in an email. And secondly, marketers want to drive traffic to their website to make ad revenue and sell subscriptions to their magazines. An example of such an article for a real newsletter can be seen in Figure 3.2. This snippet has very little text, making extracting topics using many existing topic modelling techniques more difficult [22]. Therefore, getting the full article texts from the websites was necessary.

3.1.4. Scraping

The first step of scraping the content from the publisher's website is to get all the URLs we want to scrape. For this step, having some context about the content is important. There are multiple types of content; however, we are only interested in the newsletter content for this research. This is because we know that all content records of this type have a backing article on the internet. This was considered when writing the SQL query to extract all the URLs from the database. We end up with a list of URLs that need to be scraped. The scraping was done using a Python script, which made an HTTP request to the URL and parsed the HTML. Getting the article text (excluding all the clutter of the web page) was simplified because all websites from this publisher have the same HTML layout, which made writing a script for this task trivial. After all these steps, we end up with the file described in Table 3.1a, containing the additional scraped information from the website. We can use this additional information in our analysis and feature extraction to analyse user interests.

3.2. Preliminary analysis

To answer the first research question: "Which features can be extracted to model behaviour patterns and interests?" we perform an exploratory analysis of contact results. We divide the analysis into two parts; First, we create an overview of how the data is distributed. If any outliers need to be excluded or if there is a bias in certain data points. This information can be used to clean up the data before extracting the



Dit zegt de wetenschap over intermittent fasting

Urenlang op een houtje bijten en je daarna volvreten. Intermittent fasting is een populaire manier van afvallen. Maar werkt zo'n tijdelijk eetverbod beter dan 'normaal' calorieën beperken?

Figure 3.2: An example of an article in a newsletter email

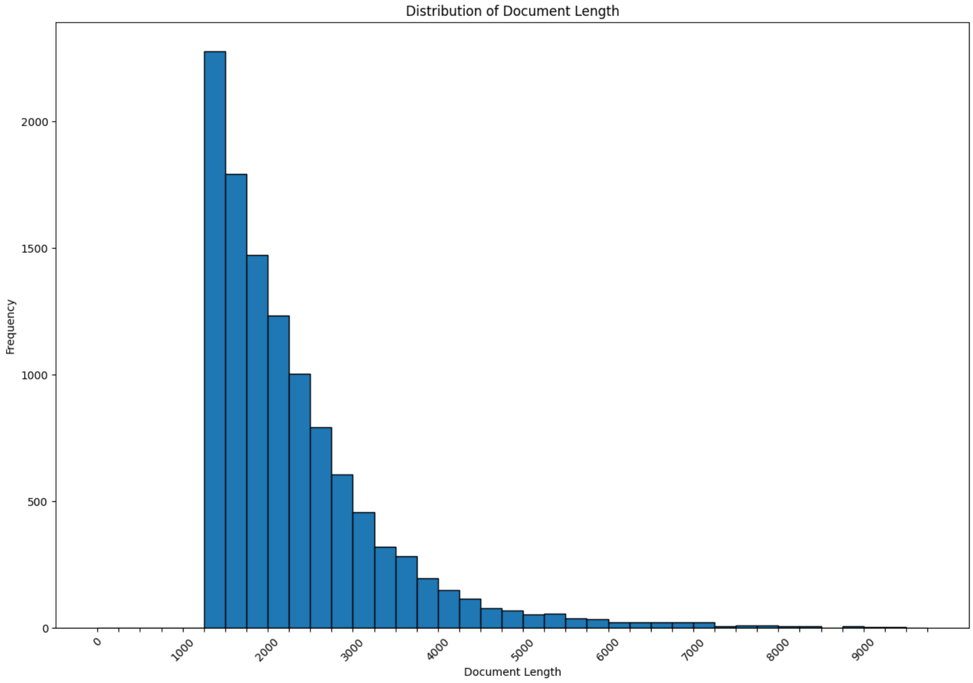


Figure 3.3: The distribution of the number of words in the document. The x-axis is the number of words in a document, and the y-axis is the number of documents in that range.

Category	Sent	Open	Ratio
gezondheid	2,368,621	786,275	33.2
geschiedenis	1,767,557	679,249	38.4
psychologie	1,678,575	629,620	37.5
dieren	1,490,915	428,911	28.8
cultuur	1,477,668	479,206	32.4
voeding	1,007,647	320,703	31.8
economie	371,402	128,704	34.7
lifestyle	367,537	112,954	30.7
sport	351,194	96,176	27.4
sterrenkunde	324,844	122,864	37.8
wetenschap	266,380	51,943	19.5
politiek	238,221	69,333	29.1
vervoer	215,470	59,724	27.7
milieu	214,086	46,551	21.7
planten	199,330	66,919	33.6
taal	198,510	56,647	28.5
technologie	195,174	37,715	19.3
geografie	98,496	32,076	32.6
misdaad	56,485	6,850	12.1
geologie	43,212	18,617	43.1
ruimtevaart	32,595	14,692	45.1

Table 3.1: Category statistics, sorted by the number of sent results

features. Secondly, we explore the topics of the clicked content. The goal of exploring the topics is to find patterns which can be transformed into features.

We perform our preliminary analysis on the subset of our data that is included in the experiment, consisting of all the profiles that are in the Quest newsletter group. The different brands of Hearst are about varying topics, which are not always relevant to each other. To reduce complexity and narrow the scope of our research, we only analyse the data that belongs to the Quest brand.

3.2.1. Sent, open and click behaviour analysis

To start our analysis we first look at the action counts and ratios. There are three defined actions: sent, open and click. Every Basedriver ID has one sent action. Every time a user opens a mail, this gets registered as a separate open action. Every click is registered as a click action. Multiple clicks on the same link get registered as separate click actions. For this analysis, the duplicate open and click actions are deduplicated, only retaining the first registered action.Figure 3.4 shows a histogram of the number of actions per profile. The duplicate click and open actions for a Basedriver ID have been removed because we want to calculate the ratios per sent and opened email. When duplicate actions are retained, these ratios become meaningless.

As is to be expected, the open rate of most profiles is low. There are some profiles which a perfect open rate. This can be because of two reasons: the user is using an email provider which opens all email by default. Or the user has only received a very small amount of mails and opened all of them.

The distributions of sent, open and click action counts in Figure 3.4 encompasses the entire dataset from 2019 up to and including 2024. However, there might be profiles that have only been receiving and interacting with mails since the last year. Or profiles that have not interacted with emails over the past year. Figure 3.5 shows the cumulative distribution of the date of the first and last click or open action. Because the open rate is generally low, the number of profiles that recently opened an email decreases rapidly. The last-click plot does not follow the same trend. Because only the first occurrence of duplicate actions is retained, reopening an already opened email is not counted as the last open action. The same applies to clicking on the same link. This means that users regularly returned to emails they previously opened and clicked on a link they had not clicked.

3.2.2. Sent and clicked article categories analysis

Analysing the sent, opened and clicked categories gives us insights into how often they are sent and how they perform. It will also help us decide on a strategy of modelling user interests such that we can extract features from this data. First, we analyse the sent and opened category counts. Second, we analyse the clicked categories. Lastly, we discuss how these results can be used to extract features.

The categories of every brand are divided into categories and sub-categories. Quest has 4 categories: ’maatschappij’ (society), ’mens’ (human), ’natuur’ (nature) and ’tech’. Every category has several more

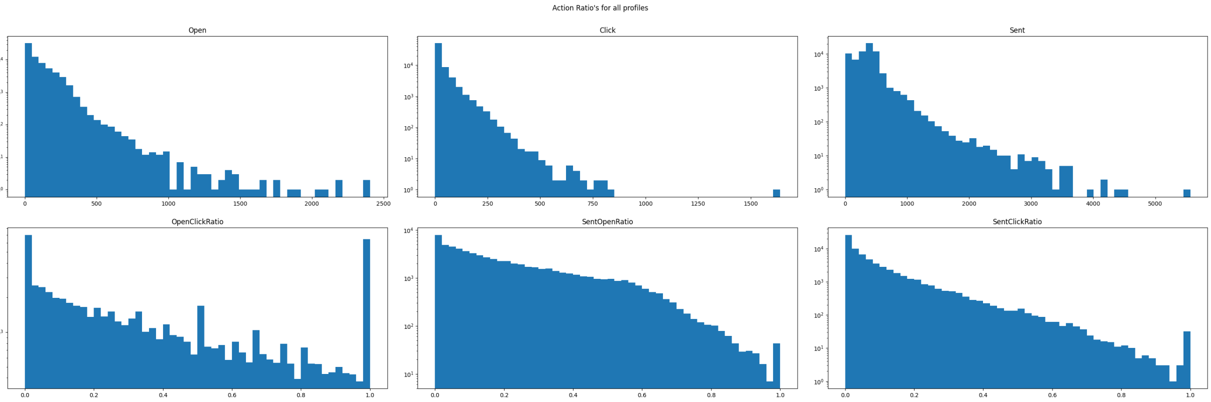


Figure 3.4: Top: histogram of a number of sent, open and click actions per profile, the y-axis is a logarithmic scale. Bottom: histogram of the open/click, sent/open and sent/click ratios

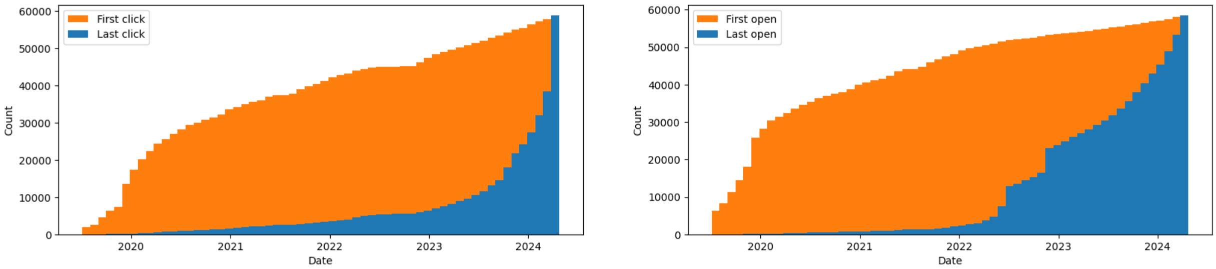


Figure 3.5: Histogram showing the number of first (left) and last (right) opens per given date

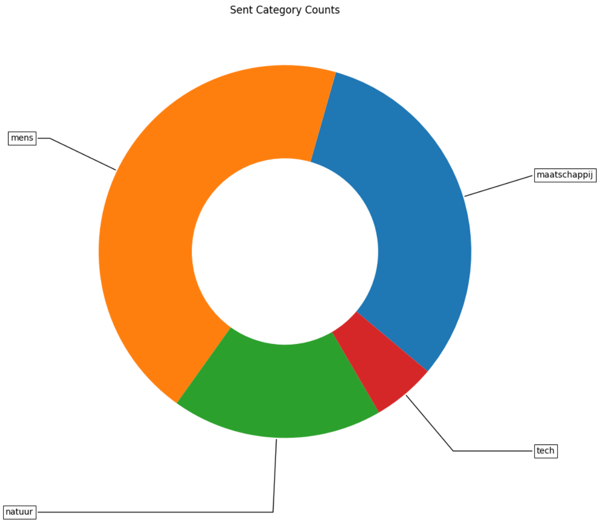


Figure 3.6: The distribution of the main categories of all sent results

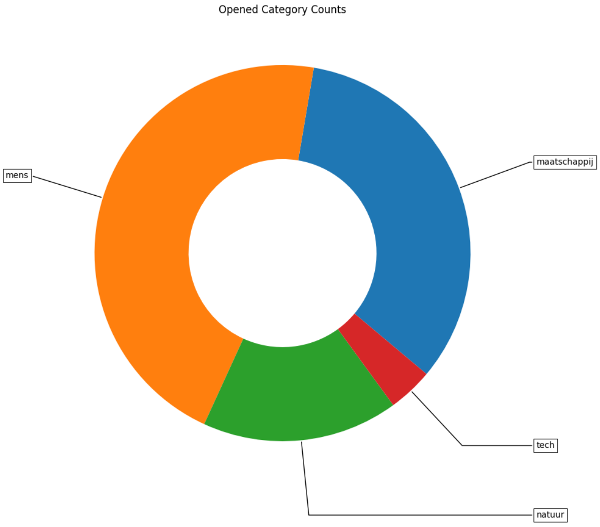


Figure 3.7: The distribution of the main categories of all open results

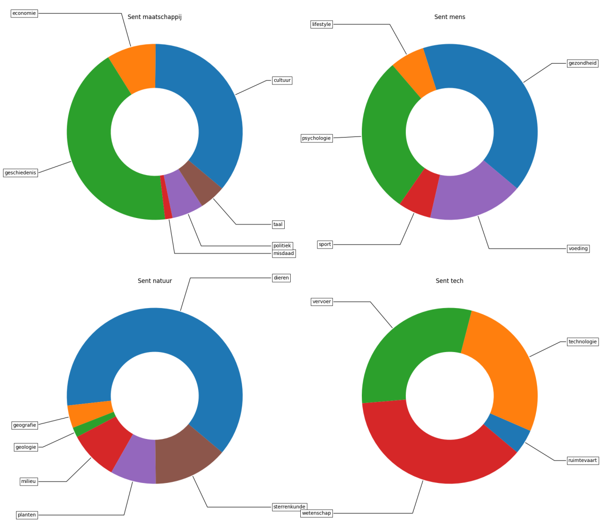


Figure 3.8: The distribution of the sub categories of all sent results

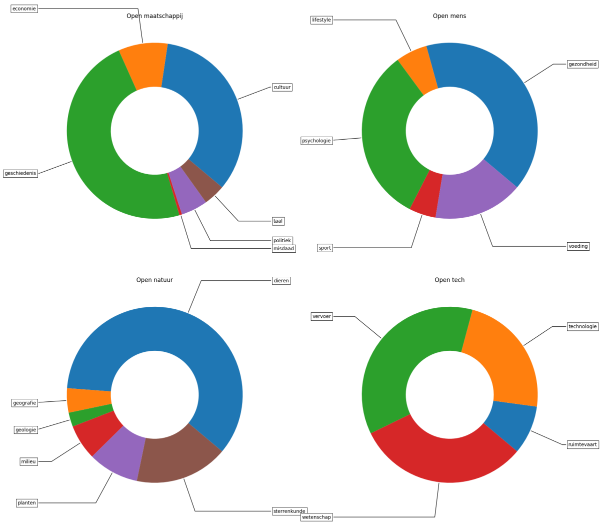


Figure 3.9: The distribution of the sub categories of all open results

specific sub-categories. Figure 3.6 shows how the categories are divided over the total number of sent emails. There is a clear preference by the marketers for the categories 'mens' and 'maatschappij'. This preference is also reflected in the number of opens on these categories, which have a similar distribution. This pattern is similarly reflected in the respective sub-categories. Readers clearly prefer these categories of which the marketers are aware. Opens on these categories are also driven by the subject line that was written with the purpose of enticing a user to click. Because our goal is to model the interests of the users, extracting features from the sent and open data is not the optimal solution. Experimentation with feature extraction on the sent and open categories shows this.

Category clicks give more information on a user's interests. A user clicks on an article after reading the headline and a snippet of the article's content. Thus, we argue that a clicked article's category better represents a user's interests than an opened article's category. Looking at figure 3.10, we observe a similar trend to those seen in figure 3.6 and 3.7; The 'mens' category is even more dominant when looking at category clicks.

For the user interests model, looking at the main categories is too broad. Therefore, we analyse the sub-categories. By counting the number of clicks per category for every user, we rank the least clicked to the most clicked category as a percentile of the total clicks. Figure 3.12 shows that 'gezondheid' is ranked the highest across all profiles. However, the figure also shows other highly ranked categories. A marketer will always try to target the group interested in 'gezondheid', as this will give the highest engagement based on the data. Ideally, a marketeer would target each interest group individually, maximising the chance of eliciting engagement.

3.3. Features

To segment the user base into user groups we look for traits that model user behaviour and user interests. Combined, we can look for groups within the total user base which have similar traits. We are modelling two types of traits: behaviour and interests. Behaviour traits model the way a user interacts with the newsletters. Grouping together people with similar interaction patterns makes it easier to compare behaviour on a group level. The second type of trait concerns interests; the interests of a user are one of the most important factors to consider when attempting to increase engagement. By understanding the interests of users, they can be grouped together into groups with similar interests. The interests of a user group can be leveraged to target the group with the content they are most likely interested in based on previous interactions, increasing the change of prompting engagement from users in the group. In this section, we discuss the feature selection process: how and why the features were selected. We also go into more detail on the representation of the features and why this representation was chosen.

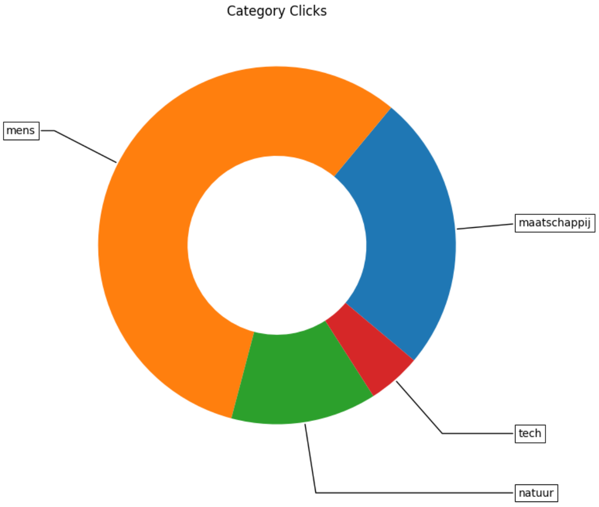


Figure 3.10: Pie-chart showing the main category distribution across all clicks

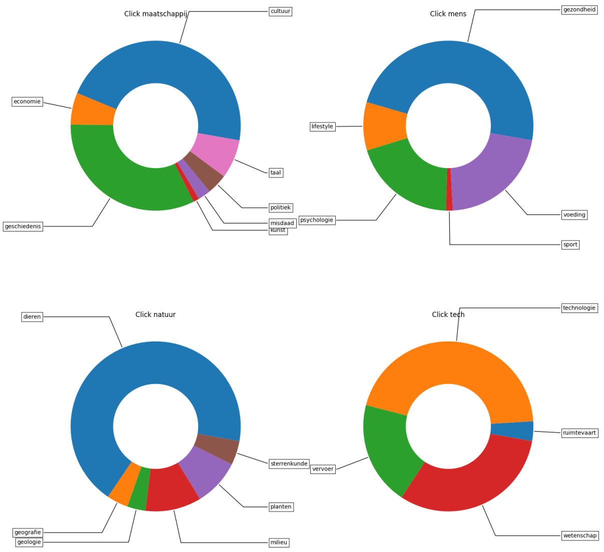


Figure 3.11: Pie-chart showing the sub category distribution across all clicks

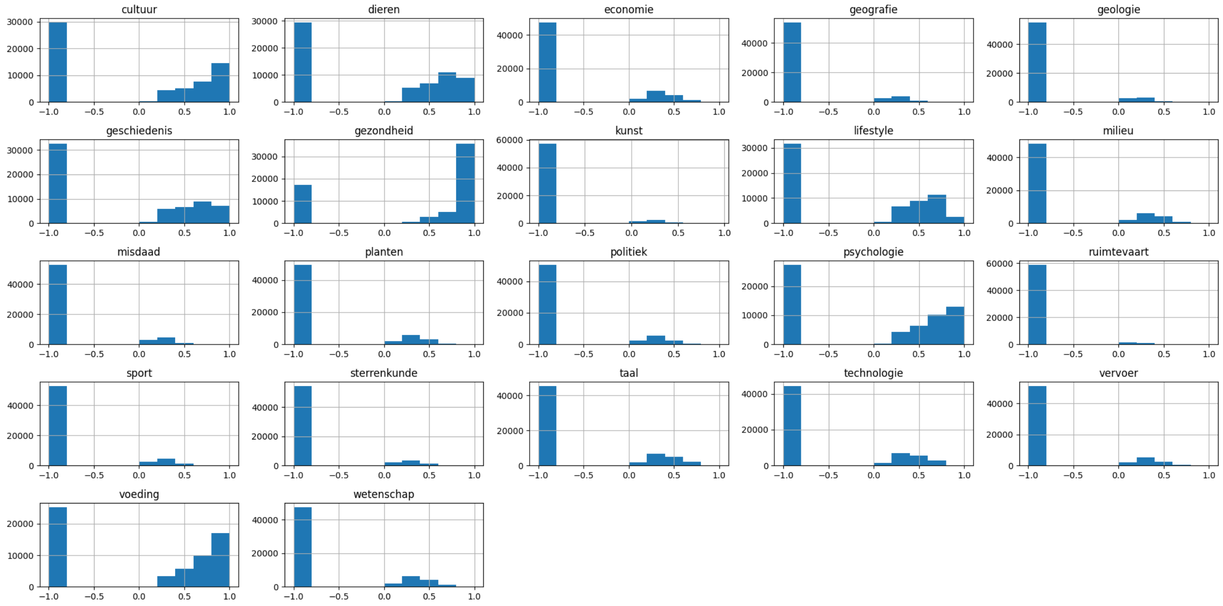


Figure 3.12: Histogram of the category ranks per profile. The ranking is done based on the number of clicks on each category, where the most clicked category gets the highest rank. The ranks are in percentiles

3.3.1. Feature selection

The goal of user features is to model the behaviour of a user in a single vector. Before creating the features vector, we must first decide what user behaviour we want to model and what is possible with the data that is at hand. We have 2 types of data available from which we can extract features: contact results and content. We focus on the contact results for now, as that is where all the features that were used were extracted from. The contact results contain all the emails that users received, as well as any actions, like a click or an open. It is also possible to extract the location of a click, such that we can compare the order in which different users click on articles in emails. From this file, we can also extract the URLs that were clicked, from which high-level topics of articles can be extracted.

Action counts

The first feature we investigate is the action counts. We calculate the action counts for the 3 most common actions: sent, open, and click. The other possible actions, like conversion or bounced, appear at a negligible rate compared to the 3 most common actions; thus, they are left out of the count completely. The counts per action per profile and their frequency can be seen in figure 3.13. The frequency of action counts rapidly decreases as the count gets higher. The outliers, which have very high action counts, have been excluded from both the graph and the entire feature extraction. These profiles have no value to us as they can't be grouped with any other profiles as the action counts are so high. In addition, many of these profiles have a very high open-and-click ratio. These profiles most likely belong to bots, or the user is using an email client which opens every email that comes in (Apple mail started doing this since iOS 15[1]). Either way, these profiles don't have any value.

The remaining profiles get a Clicks, Opens and Sent feature, which represents the number of unique actions they performed of that type. For example, if a user clicks multiple times in an email, only 1 click is counted. The same is true for when a user opens a mail multiple times. This is because this feature aims to measure how engaged a user is. If the actions were not deduplicated, this feature would be meaningless, as there is no way to measure how many emails the user engaged. However, the number of clicks on individual emails can be used for another feature.

Unique click counts

The number of unique clicks in individual emails can give us more insights into the reading behaviour of a user. If a user structurally clicks on one article, this could indicate that the user has very narrow preferences. On the other hand, if a user structurally clicks on more articles in an email, this could indicate a wide preference. The number of times a user clicks on an email could also be due to how people consume online media; one might binge-read while another reads one article and does something else. Either way, the number of clicks in an email can tell us a lot about what kind of reader the user is, which makes this a valuable feature.

This feature is calculated using the BaseDriverId. Every email that is sent has its own BaseDriverId. All actions performed on this email are linked by the BaseDriverId. We are interested in the number of unique clicks in an email for this feature. Thus, multiple clicks on the same link do not count as multiple unique clicks in the email. This process is done for every profile and every email. Every profile gets an

[1] https://www.basedriver.com/e-mailmarketing/forse-stijging-openingsratios-na-ios-15-update/

Feature	Type	Range
Clicks	Integer	0 to infinity
Opens	Integer	0 to infinity
Sent	Integer	0 to infinity
UniqueClickCounts	List of integers	List can be of length 0 to infinity. Values are bounded by the number of articles in a single mail
Categories	Double	0 to 1
ClickLinearity	Double	0 to 1

Table 3.2: Features extracted from user click data

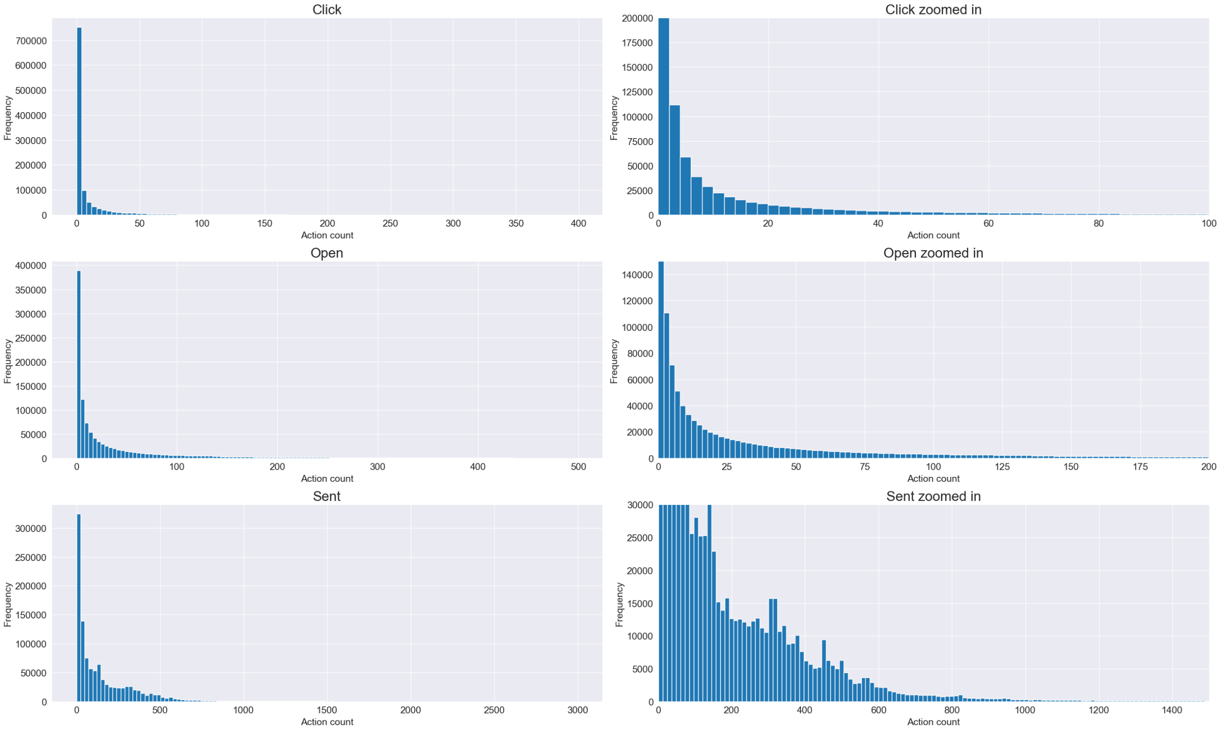


Figure 3.13: Action counts and their frequency

array with the number of unique clicks per email. For example, if a user had 1 unique click on the first email, 3 unique clicks on the second, and 2 on the third, the array would look like this: [1, 3, 2].

3.3.2. Click linearity

In addition to the number of clicks, the order of clicks within an email can also tell us much about a user's behaviour. For example, a user who clicks through the email linearly, from top to bottom, or a user who reads the articles that interest them first, returning to other articles later. We want to model this behaviour in a way that can be expressed in a single number such that it fits into the feature vector and can be used in a pre-defined distance function, such as Euclidean distance. We chose the Pearson Correlation between a strictly linear and actual clicking order to express this behaviour in a single number. The top 2 plots in figure 3.14 show examples of a high correlation and a low correlation. The red line shows the strictly linear click order that is being compared. The blue dots show the clicks and their location in the email. The x-axis represents the order in increasing time. These plots show what happens to the correlation value as the order of the clicks changes, where a less linear order of clicks gives a lower value for the Pearson Correlation (r). We also remove sequential clicks on the same location. There are several reasons for this: firstly, it happens regularly that two clicks are registered closely after each other. This could be an error or people simply clicking twice. However, this can severely skew the calculation, as is visible in the bottom two plot in figure 3.14. Removing the sequential clicks impacts the r-value significantly. Secondly, when modelling the click linearity, it is not important how many times a user clicks on a single location sequentially. We only care about the order in which a user goes through a newsletter, so removing sequential clicks is logical. To get the final value for the feature vector, we take the average correlation for all profile newsletters.

3.3.3. Categories

The previous features concern modelling user behaviour, which addresses our first research question. To address the second research question, we need to model users' interests. As discussed in section 3.2.2, we can rank the categories based on the number of clicks. Due to the number of distinct categories, using the ranking as-is in combination with the behaviour features leads to an imbalance in the weight of the features. In addition, a newsletter generally does not contain more than 6 articles. Thus, a maximum of 6

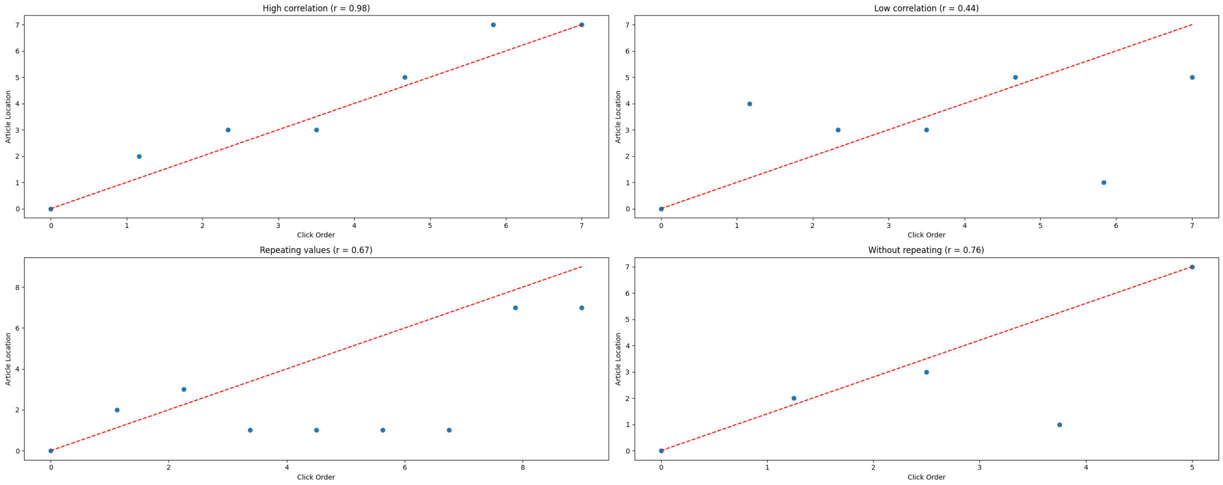


Figure 3.14: Plots of click orders and their Pearson correlation. The red line illustrates the trend the clicks need to follow in order to have a correlation of 1.0

distinct categories are present in the newsletter. Instead of considering every category when creating a feature vector, we create a feature vector for every newsletter, only considering the newsletter categories.

Rankings are converted to percentile scores. This serves two purposes: standardizing the data to the 0,1 range and encoding the relative frequency of the clicks. If a profile has no clicks on a category, we assign rank -1. Assigning rank 0 would create a very small distance between a profile with at least one click and one without on a specific category. By assigning rank -1, we prevent profiles with and without any clicks being clustered together. Reranking the categories for every newsletter will not impact the order of the categories. However, it will impact the percentile ranking score, creating tighter category clusters.

3.4. Clustering

The extracted traits from the historical clicking behaviour tell us a lot about an individual user. However, personalising on an individual level is not the most efficient. Instead of looking at individuals, we can instead group users into user groups. By grouping users, we can extract more info from the group as a whole based on the information of the individual users. In addition, by creating user groups, it is easier for marketers to keep oversight of the different subject lines that are sent out. We create user groups by clustering based on the extracted user traits features. The goal of clustering is to find similar rows (profiles) and group them together. Many different algorithms can achieve this. However, not all of them are suitable. Which algorithm is most suitable depends on the data and the use case. Before picking an algorithm, we must look at the features to see how the data is distributed. Based on this analysis, we can then make an informed decision about which algorithm best suits the use case. First, we will have a look at the feature vector that we have and how the features are distributed. We then discuss how we transform the features into a format that we can use in the clustering algorithm. Lastly, we explain the choice of clustering algorithm and a procedure for selecting the parameters.

3.4.1. Feature vector

The best-suited clustering algorithm is dependent on the data and the type of clustering we want to achieve. An example of a feature vector can be seen in table 3.3. Using a boxplot, we can visualize the distribution of the individual features. Figure 3.15 shows the distribution for the action counts. The boxplot shows that the data mainly concentrates on the lower values, with several outliers in the 95th percentile.

Feature vector pre-processing

Clustering algorithms rely on distance metrics, making them highly responsive to the scale of feature values. Since the features we extracted consist of raw counts, the range of the different features can vary in the hundreds. Therefore, the feature vector must be scaled to give each feature (roughly) equal weight. The feature vector contains 4 types of features: action counts, unique click counts, click linearity

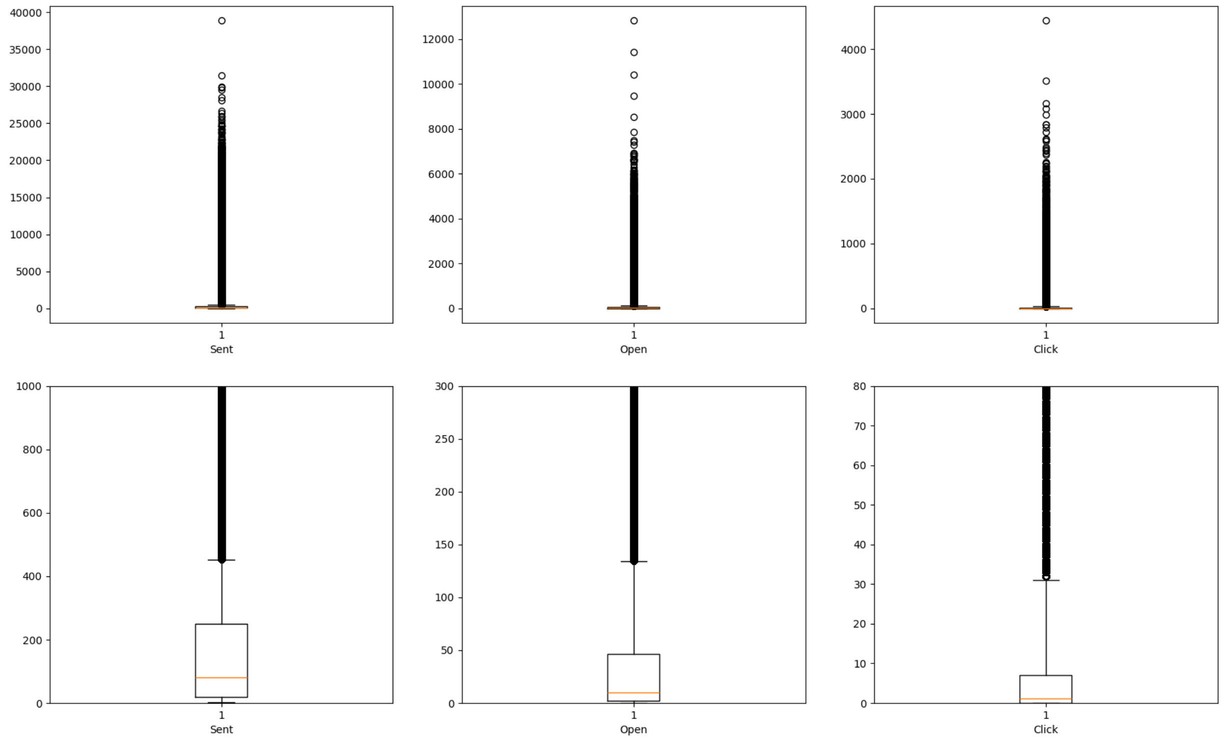


Figure 3.15: Boxplot of the action counts in the entire dataset. The top whisker is drawn at the 95th percentile boundary.

and category click counts. The action counts must be scaled, and the category counts must be ranked. The unique click count and click linearity arrays must be transformed into individual numeric columns.

The action counts are scaled using a Robust Scaler. The action counts are concentrated around the lower values and contain extreme outsider. Therefore, we scale using a robust scaler without centering the data in the 0.25 to 0.75 quantile range. This scales the majority of the values into the same range as the other features, while retaining the relatively far distance to the outliers.

The unique click counts array can have a wide range of lengths. Therefore, it is not feasible to map the array elements to columns, as this would massively inflate the size of the feature vector. Instead, the array is transformed into two columns: the standard deviation and the mean. These two values represent the distribution of click counts and can be easily compared by a distance function. There is no need to separately represent the array's length, as this is already implicitly encoded in the Click column.

The click linearity array is an array of Pearson correlation coefficients. By definition, the Pearson correlation coefficient ranges [-1 ,1] [23]. Thus, calculating the mean and standard deviation yields values ranged [-1, 1] and [0, 1] respectively. This falls within the range of the other features so there is no need to scale further.

The categories feature vector has 22 columns containing the number of clicks on that category. To increase clustering performance we remove all categories that are not in the newsletter we are creating the clusters for. Since the other categories are not represented in the newsletter, there is no added value to clustering based on them.

ProfileId	Click	Open	Sent	UniqueClickCounts	ClickLinearity	cultuur	dieren	...
2223339	5	16	38	[1, 3, 1]	[1.0, -1.0, 0.5]	2	1	...

Table 3.3: Example of a raw feature vector for a single profile

ProfileId	Click	Open	Sent	ClickCountStd	ClickCountMean	Cat1	Cat2
2223339	0.2	0.33	0.22	0.748	1.8	0.23	0.77

Table 3.4: The normalized form of the feature vector in table 3.3

3.4.2. Algorithm choice

Traditional clustering algorithms can be divided into several categories as described in 'A Comprehensive Survey of Clustering Algorithms' [24]. An overview of the different types of algorithms can be seen in table 3.5. Density-based algorithms, and DBSCAN in particular, suit the data very well for several reasons:

- There is no pre-defined number of clusters.
- Due to the nature of the features, the clusters can have any arbitrary shape.
- The data is noisy with outliers.
- The dimensionality of the data is low.

Parameter choice

DBSCAN has three parameters: distance function, *MinPts* and ϵ. The choice of parameters will heavily influence the clusters that are found. DBSCAN is extremely sensitive to the ϵ parameter; this defines the maximum distance between two for them to be considered in the same cluster [25]. We must decide which distance function to use before deciding on the ϵ. Since the features are of different types, the Manhattan Distance is the most suited. Intuitively, the Manhattan Distance also best represents the distance between two profiles, comparing and weighing every feature evenly. In addition, because most of the features are ranged [0, 1], using Euclidean distance, for example, would lead to much smaller distances. This makes picking ϵ much harder.

The best way to select ϵ is to plot a k-distance graph and look for a "valley", "knee", or "elbow" [26]. Figure 3.16 shows the k-distance graph for the profile selection of a newsletter. Smaller values of k show a clear valley between 0.2 and 0.4. However, we also see that almost 75% of the profiles fall within this range. This means that if we pick a min-cluster size of 50 for example, almost all profiles will fall within a single cluster. When we pick larger cluster sizes, we see that the valley becomes less prominent. The graph also shifts upwards, implying that for larger min-cluster values, we preferably select ϵ between 0.4 and 0.6. Based on the specific feature values, this number might fluctuate. To select a final ϵ, the parameter must be fine-tuned by observing clustering results and adjusting accordingly.

For the selection of the *MinPts* parameter, there exists a rule of thumb: $MinPts = 2 * D$ where D is the dimension [25]. This rule cannot be applied to very noisy data or a large dataset [26]. For example, creating clusters for a newsletter of 80.000 recipients and a feature vector of size 8 will result in many smaller clusters, while we would like bigger and denser clusters. However, setting $MinPts = 2 * D$ too large will result in many values being labelled as noise and fewer clusters. As with ϵ, the parameter must be fine-tuned by observing clustering results and adjusting accordingly.

Category	Typical algorithm
Clustering algorithm based on partition	K-means, K-medoids, PAM, CLARA, CLARANS
Clustering algorithm based on hierarchy	BIRCH, CURE, ROCK, Chameleon
Clustering algorithm based on fuzzy theory	FCM, FCS, MM
Clustering algorithm based on distribution	DBCLASD, GMM
Clustering algorithm based on density	DBSCAN, OPTICS, Mean-shift
Clustering algorithm based on graph theory	CLICK, MST
Clustering algorithm based on grid	STING, CLIQUE
Clustering algorithm based on fractal theory	FC
Clustering algorithm based on model	COBWEB, GMM, SOM, ART

Table 3.5: Overview of the types of traditional clustering algorithms, as outlined in [24].

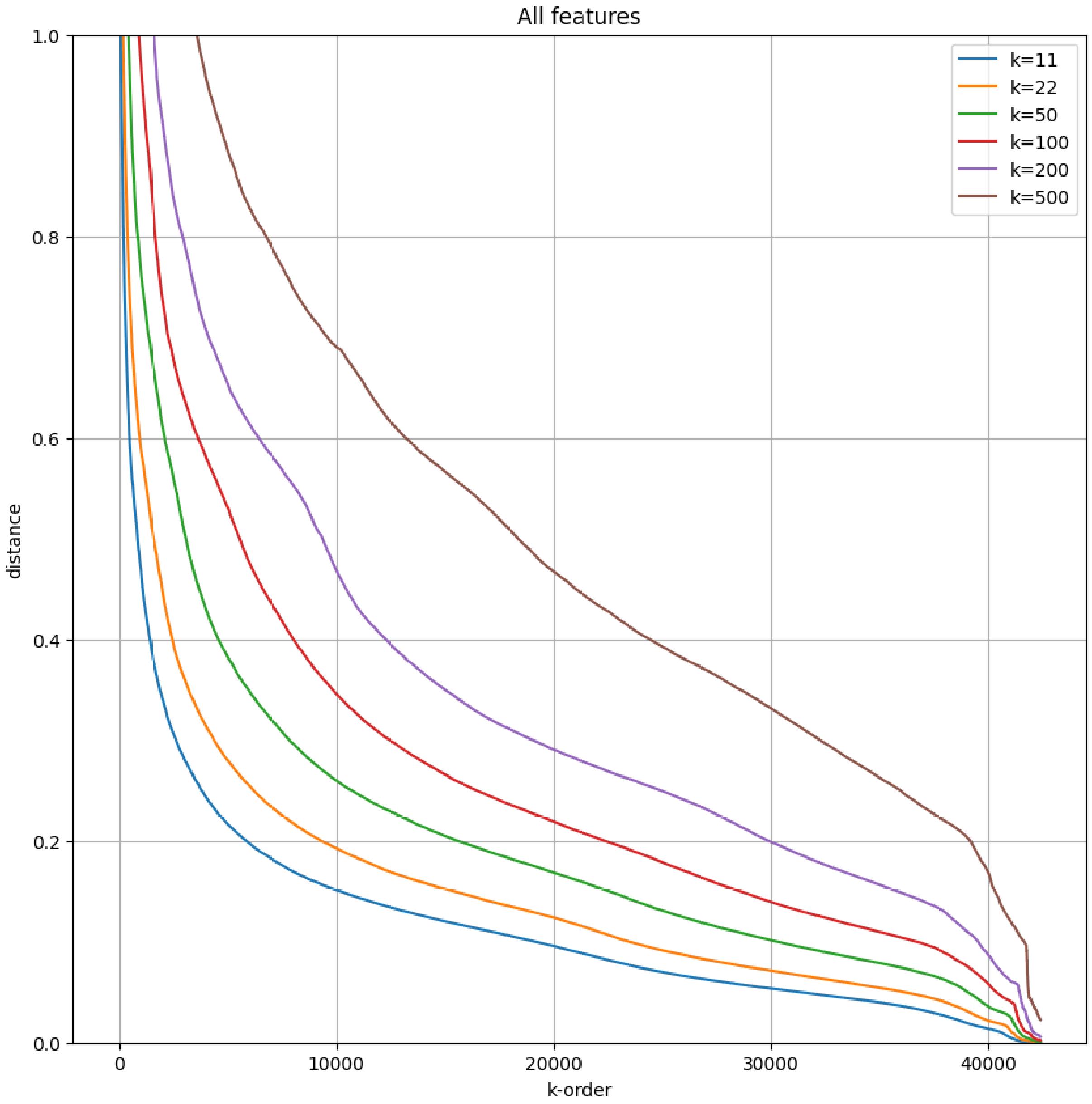


Figure 3.16: K-Distance graph for a subset of the profiles

3.5. Subject personalisation

Our final goal is to increase newsletter engagement. To this end, we have analysed users' behaviour and interests and created user groups based on this information. To help answer our last research question, we must devise a strategy for generating a subject line that leverages the information inferred from the user group. The user groups contain information about the behaviour and interests of the users. From these two, the interests of the user group can be leveraged to create a subject line that, based on what we know from the user groups, is most likely to interest the users in the group. A subject line that addresses the main interests of a user is more likely to prompt engagement.

The user's interests are represented by the number of clicks they have on the article categories. To personalise, we leverage this information to select the article in the newsletter that the users in the cluster are most likely to be interested in. To elicit an open from the users, we create a subject line referring to this article. The main goal of the personalisation is to highlight the article that the user is most interested in based on the user model. We do not perform further personalisation of generation steps because that is outside the scope of this work.

Implementation

Due to how the clustering and ranking of the categories work, we must devise a strategy for selecting the dominant category from a cluster. We do this by summing the ranks of the categories; this gives the total rank of every category in the cluster. We then take the category with the highest score to represent the interest category of that cluster. Using this category, we select the article from the newsletter that fits this category.

To generate the subject line, we use a large language model (LLM). LLMs have seen significant development over the past years, and their performance on a large variety of tasks has been steadily improved recently due to the large amount of funding and research that has gone into their development [27, 28]. LLMs provide us with a powerful which we can use to generate subject lines without having to build and train a model ourselves. Because LLMs work using prompts, we can describe the subject line generation task using human-readable language, together with any constraints. For example, it is important for a subject line to not be too long, we can easily describe this in the prompt.

There are several LLMs we can choose from. Ultimately, the choice of LLM comes down to preference and ease of use for our use case. Since the focus of our work is on the modelling and segmenting of the users, extensively comparing the performance of different LLMs is out of the scope of this work. In addition, it is non-trivial to compare the performance of LLMs on a specific task such as subject line generation, as the quality of a subject line is ultimately subjective. To generate our subject lines, we chose Google Gemini[2] as it has an easy-to-use API that we can integrate into our pipeline. We used the following prompt to generate the subject line:

```
Generate a short email subject for the following article.
Article: {articles}
```

To give an example of what a prompt might look like, we give the following example:

```
Generate a short email subject for the following article.
Article: Noem het efficiënt of onsmakelijk, maar bijna iedereen plast
weleens tijdens het douchen. ... Heb jij thuis
ook een eindeloze discussie met je partner of huisgenoot,
en wil je weten wie gelijk heeft? Laat het ons weten
via redactie@quest.nl.
```

This outputs a single subject line which we can use:

```
Plassen in de Douche: Hygiënisch of Onzin?
```

A downside of using large language models is that they can have unpredictable behaviour and sometimes hallucinate information. Therefore, the prompt's output must be manually checked with the article content to confirm that it did not hallucinate. It is very important that the subject line is factually correct and that the article content lives up to the expectations set by the subject line. As we explained in the

[2] https://deepmind.google/technologies/gemini/

introduction in Chapter 1, creating clickbait subject lines is undesirable for the brand image. Subject lines which contain incorrect information set expectations for the reader, which the articles can not live up to, leading to the worsening of the brand’s reputation.

4 Experiment

To answer the last research question: ”Do personalised subject lines increase engagement?” we conducted an experiment with one of Basedriver’s clients. The main goal of this experiment is to determine if personalised subject lines based on the most clicked category lead to more clicks when compared to the usual subject line that gets written by an editor. The client in question, Hearst Netherlands, helped us run the experiment on one of their weekly newsletters for the magazine Quest. This newsletter is sent to about 97,000 recipients weekly, giving us a large sample size on which to run the experiment. This chapter first discusses the experiment’s setup and describes the experiment protocol. After which, we discuss the results of executing the experiment.

4.1. Setup and Protocol

We have discussed how we analysed the historical click data and used this analysis to extract features from which created user groups. In Section 3.5, we explained how we can leverage the information from the user groups to create personalised subject lines for every user group. The experiment aims to combine all these methods and create user groups and personalised subject lines for a single newsletter email. We can answer our last research question by sending out this newsletter with the subject lines we created using our methods and analysing the results. For this purpose, we have devised the following protocol:

- Data analysis: analyse the data and extract features for every profile.
- Clustering: cluster the profiles to obtain user groups using the features.
- Subject Personalization: per user group, create a subject line specific to that group’s interests.
- Send email: Send the newsletter to all clusters using the generated subject line.
- Collect and analyse results: collect the results and calculate metrics to analyse the impact of personalization on the users’ engagement.

4.1.1. Data analysis

To model the 97,000 recipients of the weekly newsletter of Quest based on their past interactions with newsletters, we first extract the profile IDs of the newsletter receivers from the Basedriver database. We also extract the content created by the marketers. We need the content to extract the article categories needed for creating the feature vector, as explained in Section 3.3.3. Using the profile IDs and the categories, we extract the features from historical click data; this yields all the feature vectors.

4.1.2. Clustering

To produce user groups based on the features extracted, we run DBSCAN. Since DBSCAN is very sensitive to parameter changes, we ran the clustering with different parameters and observed the results. The goal is to cluster the most profiles, preferably clustering as many profiles as possible. In addition, we want the clusters to be concentrated to prevent clustering profiles together with very different behaviour and interest patterns. Thus, we started with a low epsilon and min-samples value and slowly changed and observed the results. Table 4.2 shows the parameters that gave the best results and were ultimately chosen.

4.1.3. Subject personalization

To create the subject lines sent to every user group, we leverage the group's interests to select an article and generate a subject line. Personalization is done as described in section 3.5; namely, the most frequently clicked article category for that cluster is selected from the newsletter for that group, for which a title is generated using Google Gemini. User groups that have the same category also receive the same subject line. The focus of our experiment is to validate whether the clustering correctly identifies user groups and their interests and whether catering to these interests increases overall engagement. Therefore, how the subject lines are generated is not the focus, as this only serves as a means to test whether the interests identified by the clustering lead to higher engagement.

The generated subject lines were sent to Hearst before running the experiment, where the marketing team behind the newsletters checked them and made minor tweaks for them to be in line with their usual writing style. This mainly concerned changes to words; the structure, meaning and tone of the subject line remain the same.

4.1.4. Send email

The newsletter we chose for this experiment is sent every Friday around 5 P.M. To prevent the sending time influencing the results, we execute the experiment on the same day and time. The content and order of the email are created by the marketers, in between user groups we do not touch this.

4.1.5. Collect and analyse results

After sending the mail, we waited 7 days before collecting the contact results containing the user's interaction with the email. Based on experience and historical data, we expect most results to be in after 7 days. In addition, the next newsletter would be sent around this same time. After this collection period of 7 days, we proceed with data analysis. For this purpose, we use a variety of metrics which aim to capture the level of engagement from several perspectives. By comparing the engagement on the experimental email to previous interactions within the user groups, we can determine the effect personalisation had on the engagement. The metrics are defined as follows:

- The send/open ratio.

  Hypothesis: Personalising the subject line increases the send/open ratio.
- The open/click ratio.

  Hypothesis: Personalising the subject line increases the open/click ratio.
- Average number of clicks within an email.

  Hypothesis: Personalising the subject line leads to more clicks within emails.
- Total clicks.

  Hypothesis: Personalising the subject line leads to more total clicks.
- Response time. Defined as the time between the send time of the email and the open time.

  Hypothesis: Personalising the subject line decreases email response time.

4.2. Results

4.2.1. Features

Following the procedure described in Section 4.1, we started by extracting the profiles that will receive the newsletter. This resulted in a selection of 97,044 profiles. We extracted the behaviour and interest features based on their historical behaviour for this selection. To extract the category features, we must know the categories of the articles in the newsletter, as these will determine the category features. Before extracting the feature, we therefore first extracted the categories from the content. Using the categories, we extracted the features.

It is impossible to extract the features from profiles with no clicks on the categories in the newsletter; thus, these are excluded. After extracting the features and removing the profiles without values, we are left with 43,382 profiles and their feature vectors.

4.2.2. Clusters

We cluster the profiles using DBSCAN, for this we use the following parameters: $\varepsilon = 0.50$, $minsamples = 250$ and the L1 metric. These parameters were selected as described in Section 3.4. This results in a total of 28,719 clustered profiles distributed across 10 clusters, which can be seen in Table 4.1. The remaining profiles are labeled as 'noise' by DBSCAN, meaning that these profiles do not belong to any cluster. Looking at how the feature values are distributed across the different clusters in Figure 4.1, we can see where the clusters concentrate. Especially for the category features, the interests for the different clusters are visible. Looking at the open/click and open/sent ratio, there is a less defined border and the range of the feature values is broader.

Cluster	Category	Size
0	gezondheid	3658
1	gezondheid	16188
2	gezondheid	3471
3	gezondheid	5011
4	gezondheid	3289
5	psychologie	973
6	gezondheid	1804
7	dieren	830
8	psychologie	1036
9	psychologie	384

Table 4.1: The user groups obtained after clustering. The category describes the most frequently clicked article category for that cluster. The size is the number of unique profiles in the cluster.

Total profiles	97,044
Profiles with at least 1 click	43,382
Noise	14,994
Clustered	28,719

Table 4.2: The number of profiles we can use for clustering (with at least 1 click) and the number of profiles clustered using DBSCAN. Profiles that DBSCAN could not put into a cluster, are labeled as noise.

4.2.3. Subject lines

For every cluster, we select the category that is most likely to create engagement based on our user model. This results in the categories for every cluster as shown in Table 4.1. Most clusters have the category 'gezondheid', which is the category that the marketers predicted would be the category that most users would be interested in. We also have 3 clusters that receive a subject line about 'psychologie' and 1 that will receive a subject line about 'dieren'.

The email that was used in the experiment contained other categories, namely 'maatschappij' and 'sterrenkunde'. For the category 'sterrenkunde', there are no clusters in which this category has historically received the most clicks. The article about 'maatschappij' concerns a sponsored article, which for some clusters was the most frequent and would have been included in the personalisation. However, the company does not want to feature sponsorships in the subject line as per internal arrangements. Thus, we were forced to exclude this article from the personalisation. This leaves 3 categories for which we generate a subject line:

- gezondheid: Hoofdpijn: wat doet er dan pijn?
- psychologie: Afdwalen tijdens de Dodenherdenking? Je bent niet de enige!
- dieren: Leeuw en mier: een onverwachte vriendschap

4.2.4. Analysis

Seven days after we sent the email we collected the results and calculated all metrics. We first calculate every cluster's sent/open and open/click ratio. This is done by dividing the number of unique actions within the cluster. We compare to the historical data of the cluster, meaning we look at all previous emails for every profile in the cluster and take the average. The result of these calculations can be seen in Tables 4.3 and 4.4. For the sent/open ratio, we observe an increase for every cluster, especially for the smaller clusters we see an increase from quite a low number. However, this can also be explained by the recent

Cluster	Historical	Experiment	Difference
0	0.57	0.69	0.11
1	0.38	0.57	0.19
2	0.26	0.50	0.24
3	0.20	0.46	0.27
4	0.26	0.52	0.26
5	0.21	0.44	0.24
6	0.57	0.73	0.16
7	0.16	0.43	0.27
8	0.16	0.42	0.25
9	0.23	0.44	0.21

Table 4.3: The difference between the sent/open ratio of the experimental mail and the historical sent/open ratio of the entire cluster

Cluster	Historical	Experiment	Difference
0	0.63	0.47	-0.16
1	0.40	0.28	-0.12
2	0.22	0.15	-0.07
3	0.13	0.10	-0.03
4	0.20	0.13	-0.06
5	0.13	0.03	-0.10
6	0.58	0.42	-0.16
7	0.11	0.03	-0.09
8	0.11	0.01	-0.09
9	0.17	0.04	-0.13

Table 4.4: The difference between the open/click ratio of the experimental mail and the historical open/click ratio of the entire cluster

increase in email clients automatically opening all emails as spam protection. Since the historical also contains data from before this was done, we cannot draw any definitive conclusions from this metric.

We also observe that every cluster's open/click ratio has decreased. Looking at the clusters which did not receive a subject line regarding 'gezondheid', we see that they received little to no clicks. For the 'gezondheid' clusters, although lower, the number of clicks is still significantly higher than that of the non-'gezondheid' clusters. It is possible that the topic discussed in the subject line regarding 'gezondheid' is less engaging in general when compared to previous subject lines, leading to a general decrease in clicks. The cluster receiving the 'dieren' and 'psychologie' subject lines performed significantly worse, implying that personalising for these groups most likely did not lead to increased engagement.

Cluster	Historical	Experiment	Difference
0	1.67	1.74	0.07
1	1.31	1.27	-0.04
2	1.19	1.18	-0.01
3	1.14	1.21	0.07
4	1.16	1.22	0.06
5	1.11	1.54	0.43
6	1.54	1.66	0.12
7	1.06	1.00	-0.06
8	1.10	1.17	0.07
9	1.11	1.00	-0.11

Table 4.5: Difference between the average unique clicks of the experimental mail and the average unique clicks of the entire history of the cluster

Cluster	Historical	Experiment	Difference
0	1.64	2.33	0.68
1	1.30	1.63	0.33
2	1.18	1.52	0.34
3	1.14	1.55	0.41
4	1.16	1.52	0.36
5	1.12	2.31	1.19
6	1.52	2.24	0.72
7	1.08	1.22	0.14
8	1.12	1.17	0.05
9	1.12	1.00	-0.12

Table 4.6: Difference between the average total click of the experimental mail and the average total click of the entire history of the cluster

We also calculate the number of unique clicks and the number of total clicks. Just like with the ratios, we calculate these metrics for every profile in the cluster and take the mean. We compare this value to the mean of the cluster's historical behaviour. Looking at the unique clicks in Table 4.5, we observe one outlier, cluster 5. This cluster has a significant increase when compared to the other clusters, which barely show any movement. To explain this increase, we look at the actual clicks of cluster 5. There are only 13 users who interacted with the email, of which most have 1 or 2 clicks. However, there is one outlier who clicked 5 times. Because of the small sample size, this has a large impact on the mean. Therefore, we can discard these results and conclude that personalisation is unlikely to increase the average number of unique clicks.

We calculate the total clicks similarly to the unique clicks, comparing the historical means to the experimental means per cluster. The results of this can be seen in Table 4.6. Here, we see more diverse changes compared to the unique clicks. However, when looking at the data of the clusters which show an increase, we see that this is mostly due to outliers. These outliers are also present in the historical data. However, because this concerns multiple email, these outliers weigh less in the mean. Therefore, we conclude that it is unlikely that personalisation leads to an increase in the total clicks.

Cluster	Historical	Experiment	Difference
0	3601.0	1688.0	-1913.0
1	2196.0	1405.0	-791.0
2	2000.0	1185.0	-815.0
3	2099.0	1301.0	-798.0
4	1642.0	1175.0	-467.0
5	1850.0	1364.0	-486.0
6	2952.0	1767.0	-1185.0
7	1747.0	958.0	-789.0
8	1757.0	796.0	-961.0
9	1788.0	780.0	-1008.0

Table 4.7: Difference between the average response time of the experimental mail and the average response time of the entire history of the cluster. The values are in minutes.

Finally, we calculate the average response time. This is defined as time between the send time an email and the first click result that is recorded. We look at the click result, as we can not trust the open time due to email clients opening mails automatically. As with the previous metrics, we compare the mean of the historical data to the mean of the experimental data. The results in Table 4.7 show that all clusters have a smaller mean response time, with varying differences. However, the standard deviation of the response times within the cluster is very big, approaching the value of the mean. Because the historical data of the response time contains many outliers, we get a high mean value. In addition, we can not draw conclusions from the smaller clusters, as the sample size is very small.

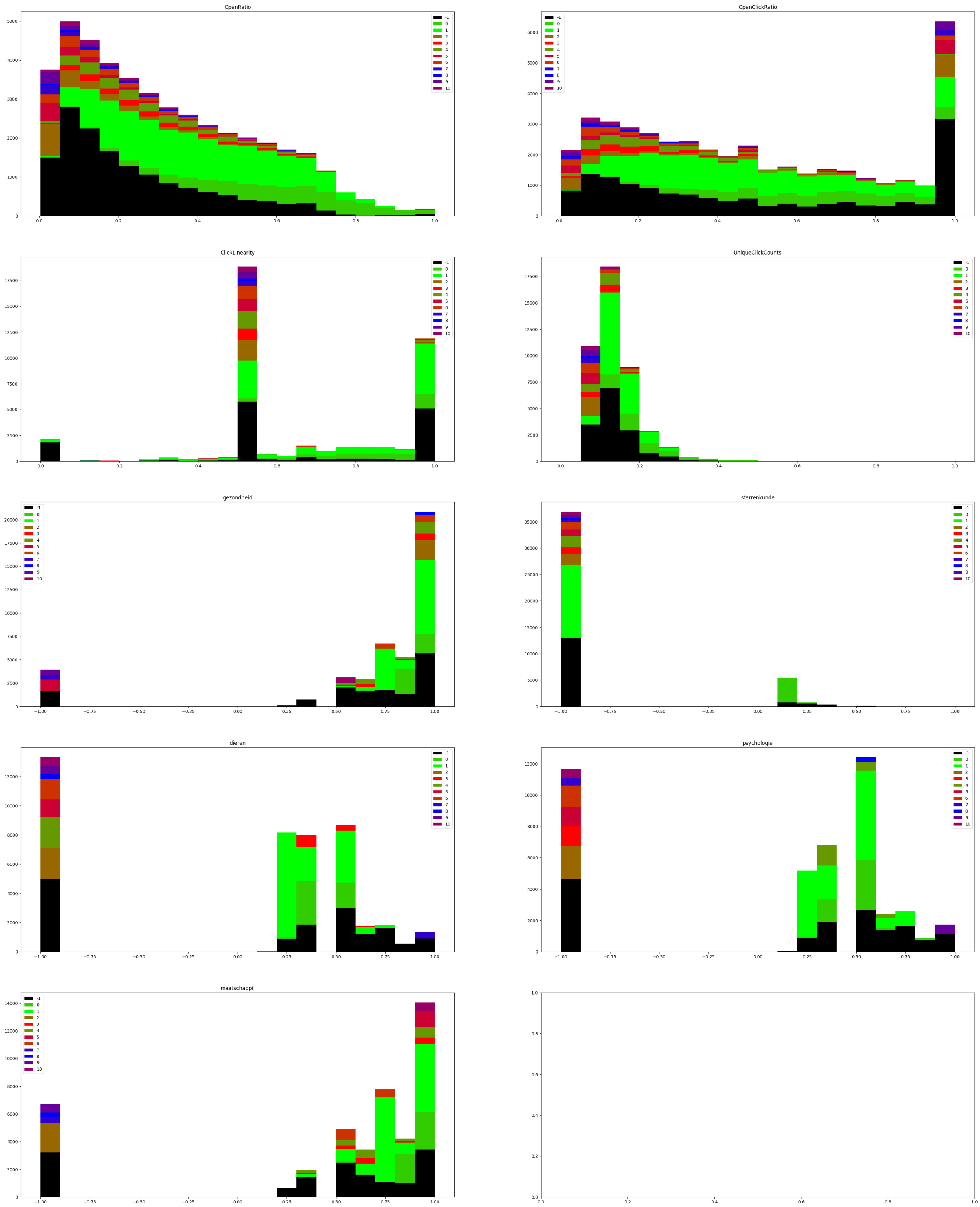


Figure 4.1: The distribution of the feature values of the profiles in the experiment newsletter. The colours indicate the different clusters that were created.

5 Discussion

5.1. Limitations

5.1.1. Analysis and modelling

During our analysis, we focused on the clicked articles to determine the interests of users. Although this has shown to be effective in modelling the interests of already engaged users, we have shown that this does not allow for an overall increase in engagement across all users. This is because of several reasons. Firstly, for a user to click on an article, they must open the mail containing said article. Because marketers perform analysis on the performance of their subject lines already and because of their experience, they are aware of the subjects that appeal to the broadest possible audience and tailor the subject lines accordingly. This creates a bias toward certain topics that marketers know work and interest the broadest possible audience. Because users first need to be drawn in by the subject line, they are unaware of the other topics presented in a newsletter. While the topic in the subject line might not appeal to them, another topic in the newsletter might. However, because the user is unaware of this, getting to know these interests is impossible. Because of the bias towards certain topics in the subject line, it is very difficult to model users' interests who gravitate towards underrepresented topics in the existing subject lines.

Secondly, by looking at clicked articles we are implicitly modelling users who already have at least a moderate level of engagement. The total user base can be divided into three groups: users who don not engage at all, users who sometimes engage and users who always engage. Trying to Target the first and last group will have little effect, as, by nature, these users' behaviour is hard to influence. The second group is where a personalisation strategy will have the most effect. However, by only looking at clicked articles when modelling interests, we mostly cater to the most engaged group, as this group will have the most data. To try and engage this group of users, we could look at the topics that receive the least amount of engagement and try engaging them by sending subject lines referencing topics that are generally not referenced. This combats the previously discussed topic bias by introducing a more diverse set of topics, possibly exposing more defined user groups instead of the monotone groups we currently see. By probing this under-engaged group with different topics, we can more easily identify the topics that will engage these users.

Lastly, for our analysis, we looked at all the clicks over the past 5 years without considering the time dimension; this presents two problems: certain topics might be more relevant at a specific time; we call these short-term interests. For example, a war or sports event creates a short peak of interest for a large group, while not everyone is interested in these topics when there are no world events. Over the long term, when a user has a lot of clicks, these temporary interests will have less weight as the number of clicks on other articles far outnumbers these temporary interests. However, for users who don't have a lot of clicks, this makes it difficult to model their long-term interests. In addition to the lack of differentiation between short and long-term interests, a user's long-term interests can change over time. For example, a user might start going to the gym and watching their health more. If a user has been subscribed for a few years, it can take a long time and many clicks before these changes start outweighing their old interests.

5.1.2. Features

The selection of features presents some limitations which negatively impact the performance of our model. Firstly, two of the features represent the sent/open and open/click ratio. Using the ratio instead of the raw counts as a feature has several advantages. For example, the values are, per definition, standardized to a 0,1 scale, which allows for easy distance calculations. However, there are also disadvantages. The biggest disadvantages are the lack of weight and loss of information. There is no way of knowing whether a ratio is based on 10 or 1000 results. Because we extracted features based on a dataset that spans 5 years of information, the number of results per profile varies a lot. Especially when comparing newer profiles to older ones.

The ratios also create a continuous scale, which is clearly visible in the top 2 plots of Figure 4.1. Because the distance between profiles is so small for these features, it is difficult to find a clear boundary for the clusters. This also shows in how this feature is distributed in the clusters, other features have more defined cluster centers, while the ratios are more spread out. Secondly, the feature vector comprises different types of features with vastly different meanings. Because of this, we had to convert the features to a common format, which can be compared using distance metrics. By converting the features, we lose information on the meaning of some features. For example, the click linearity and unique clicks features represent distributions. Preferably, we would compare these using statistical metrics to express the meaning of the difference better. However, due to computational constraints, we were unable to create custom metrics. Using a custom metric, we could have compared features in their original form using metrics best suited for comparing that type of feature. However, due to our implementation in Python, a custom metric in combination with DBSCAN was impossible. Incorrectly converting and scaling the features can impact the clustering, as profiles might be incorrectly grouped, impacting the experimental results.

5.1.3. Experiment setup

Because of the limitations in the modelling and the biases in the data, clustering resulted in user groups which mainly focused on one topic, namely the topic the marketers predicted would perform best. Although this does confirm that our model is able to come to the same conclusion as the marketers, there is little added value, as we were unable to significantly show an increase in engagement as a result of our personalisation strategy. The user groups that showed potentially different interests were very small compared to the other user groups. Preferably, we would have performed an A/B test per cluster to compare the behaviour of similar users. This would have also allowed us to perform statistical tests to determine significance. However, because engagement on emails is low in general, splitting a small cluster into even smaller groups would have resulted in sample sizes too small to show any significance. To still be able to analyse the engagement, we therefore compared the experiment results to the entire historical behaviour of the cluster. Since historical behaviour can encompass data from a wide variety and a large number of emails, we cannot draw any definitive conclusions from the results.

The experiment was performed in collaboration with Hearst. Performing an experiment with a real company and their paying customers has many benefits. For example, users are unaware they are part of an experiment, so their behaviour is not influenced. Also, it gives us a lot of real-world data to analyse and insights from the people working with the data daily. However, it also presents some limitations that we have content with. In general, marketers have a strong opinion about what works best and like to stick to a certain writing style for their subject. We generated the subject lines with a large language model, which struggled with writing subject lines in the style of the marketers. Because we are working with real customers, the marketers want to prevent sending out a subject line they do not stand behind. The subject lines were thus slightly changed by the marketers to reflect their usual writing style better. Although this does not directly impact the main focus of our study, since we are evaluating the modelling and clustering performance, it is important to keep this in mind when performing further research on this topic.

6 Conclusion

With this work, we aimed to explore how we can segment a user base into user groups based on behaviour and interests; levering this information to increase engagement by personalising the subject line based on interests extracted from the user groups. We performed this exploration in the context of a real publisher, Hearst, and one of their brands, Quest, who sent out weekly newsletters to their customers. Basedriver, the email marketing company Hearst uses to sent their email, provided us with the users historical click data.

Based on an analysis of the historical clicking behaviour of 97,000 users who receive the weekly Quest newsletter, we extracted behaviour and interest features and clustered them into user groups. Using clicked articles in every user group, we generated subject lines and explored the impact on engagement using an online experiment on the real users.

From this experiment, we came to the conclusion that the majority of the user groups converge on the same topic. We also found no significant change in engagement, even observing a decrease in engagement in some user groups. Based on these results, we conclude that personalising subject lines for user groups based on historical clicking behaviour does not lead to higher engagement. Further analysis of the clusters created in the experiment revealed biases in the categories users clicked on. We discussed the limitations of our modelling and segmentation approach, which resulted in these biases, and how this affects engagement.

6.1. Future Work

Following our discussions on the limitations of our work, we have identified several directions which future works could explore.

Alternative Approach for Modeling Interests. Our current interest modelling primarily focuses on positive engagement signals, such as clicks on articles. However, because of existing biases of marketers towards certain topics in the subject lines, users not interested in these topics never engage. This makes modelling these users using our approach almost impossible. A promising direction for future work involves analyzing the topics of subject lines that did not generate engagement. We can gain insights into their disinterests by understanding what topics users consistently ignore. This negative feedback can be used to create an alternative model, modelling topics the users are not interested in. Additionally, we can incorporate a strategy that promotes less frequently represented subjects to counteract the inherent bias towards popular topics. This approach diversifies the content exposure and helps uncover undiscovered interests that may not be immediately apparent through looking at clicked articles.

Differentiating Between Long- and Short-term Interests. User interests can be influenced by current events, with some preferences being only relevant because of world events like a war, for example. To address this, future work should focus on separately modelling long-term and short-term interests. Long-term interests could be derived from consistent engagement patterns over extended periods, while short-term interests might be identified from recent spikes in activity on specific topics. Maintaining distinct models for these two types of interests can create more nuanced and timely recommendations, enhancing user engagement.

Weighing the Age of Results. While short-term interests represent peaks, long-term interests can also change over time. By equally weighing all results over the past years, we are giving results from years ago the same weight as more recent results. This not only incorrectly models a user's current interests, but it also makes it more difficult to extract the interests from a user, as the total weight of different results you are considering is much higher. Future work should, therefore, take the age of results into account when modelling interests.

To conclude this thesis, we have shown that personalising for user groups segmented on historical clicking behaviour does not lead to higher engagement. However, following further analysis of experimental results, we provide a starting point and promising directions for future work to explore further how newsletter engagement can be improved by modelling users' behaviours.

References

[1] Jeffrey Kuiken et al. “Effective headlines of newspaper articles in a digital environment”. In: *Digital Journalism* 5.10 (2017), pp. 1300–1314.

[2] Ángela Bazaco et al. “Clickbait as a strategy of viral journalism: conceptualisation and methods”. In: *Revista Latina de Comunicación Social* 74 (2019), p. 94.

[3] Joshua M Scacco et al. “Investigating the influence of “clickbait” news headlines”. In: *Engaging News Project Report* (2016).

[4] Stephen R Porter et al. “E-mail subject lines and their effect on web survey viewing and response”. In: *Social Science Computer Review* 23.3 (2005), pp. 380–387.

[5] Natalie Sappleton et al. “Email subject lines and response rates to invitations to participate in a web survey and a face-to-face interview: the sound of silence”. In: *International Journal of Social Research Methodology* 19.5 (2016), pp. 611–622.

[6] Jaclyn Wainer et al. “Should I open this email? Inbox-level cues, curiosity and attention to email”. In: *Proceedings of the SIGCHI conference on human factors in computing systems*. 2011, pp. 3439–3448.

[7] Logan Molyneux et al. “Aggregation, clickbait and their effect on perceptions of journalistic credibility and quality”. In: *Journalism Practice* 14.4 (2020), pp. 429–446.

[8] Abid Haleem et al. “Artificial intelligence (AI) applications for marketing: A literature-based study”. In: *International Journal of Intelligent Networks* 3 (2022), pp. 119–132.

[9] Sanjeev Verma et al. “Artificial intelligence in marketing: Systematic review and future research direction”. In: *International Journal of Information Management Data Insights* 1.1 (2021), p. 100002.

[10] Christi Olson et al. “Transforming marketing with artificial intelligence”. In: *Applied Marketing Analytics* 3.4 (2018), pp. 291–297.

[11] Patrick Van Esch et al. “Artificial intelligence (AI): revolutionizing digital marketing”. In: *Australasian Marketing Journal* 29.3 (2021), pp. 199–203.

[12] M Paulo et al. “Leveraging email marketing: Using the subject line to anticipate the open rate”. In: *Expert systems with applications* 207 (2022), p. 117974.

[13] Raju Balakrishnan et al. “Learning to predict subject-line opens for large-scale email marketing”. In: *2014 IEEE International Conference on Big Data (Big Data)*. IEEE. 2014, pp. 579–584.

[14] Andreia Conceição et al. “Main factors driving the open rate of email marketing campaigns”. In: *Discovery Science: 22nd International Conference, DS 2019, Split, Croatia, October 28–30, 2019, Proceedings 22*. Springer. 2019, pp. 145–154.

[15] Xiao Luo et al. “Predictive analysis on tracking emails for targeted marketing”. In: *Discovery Science: 18th International Conference, DS 2015, Banff, AB, Canada, October 4-6, 2015. Proceedings 18*. Springer. 2015, pp. 116–130.

[16] Rui Zhang et al. “This email could save your life: Introducing the task of email subject line generation”. In: *arXiv preprint arXiv:1906.03497* (2019).

[17] John Crunk et al. “Decision support systems and artificial intelligence technologies in aid of information systems based marketing”. In: *International Management Review* 3.2 (2007), pp. 61–67.

[18] Marius Geru et al. “Using artificial intelligence on social media’s user generated content for disruptive marketing strategies in eCommerce”. In: *Economics and Applied Informatics* 24.3 (2018), pp. 5–11.

[19] Koldyshev Maxim Vladimirovich. “Future marketing in B2B segment: Integrating Artificial Intelligence into sales management”. In: *International Journal of Innovative Technologies in Economy* 4 (31) (2020).

[20] Nazirul Shovo. “Marketing with artificial intelligence and predicting consumer choice”. In: *Artificial Intelligence in Society* 1.1 (2021), pp. 6–18.

[21] Shobhana Chandra et al. “Personalization in personalized marketing: Trends and ways forward”. In: *Psychology & Marketing* 39.8 (2022), pp. 1529–1562.

[22] Rania Albalawi et al. “Using topic modeling methods for short-text data: A comparative analysis”. In: *Frontiers in artificial intelligence* 3 (2020), p. 42.

[23] Israel Cohen et al. “Pearson correlation coefficient”. In: *Noise reduction in speech processing* (2009), pp. 1–4.

[24] Dongkuan Xu et al. “A comprehensive survey of clustering algorithms”. In: *Annals of data science* 2 (2015), pp. 165–193.

[25] Jörg Sander et al. “Density-based clustering in spatial databases: The algorithm gdbscan and its applications”. In: *Data mining and knowledge discovery* 2 (1998), pp. 169–194.

[26] Erich Schubert et al. “DBSCAN revisited, revisited: why and how you should (still) use DBSCAN”. In: *ACM Transactions on Database Systems (TODS)* 42.3 (2017), pp. 1–21.

[27] Wayne Xin Zhao et al. “A survey of large language models”. In: *arXiv preprint arXiv:2303.18223* (2023).

[28] Jason Wei et al. “Emergent abilities of large language models”. In: *arXiv preprint arXiv:2206.07682* (2022).