UiT The Arctic University of Norway

Faculty of Science and Technology
Department of Computer Science

Large Language Models for Managing Online Fingerprint

Danielle Fredrikke Olaisen Vik

Master's thesis in Computer Science INF-3981 - June 2024

Supervisors

Main supervisor:	Elisavet Kozyri	UiT The Arctic University of Norway, Faculty of Science and Technology, Department of Computer Science

“Getting information from the internet is like taking a drink from a hydrant.”
–Mitchell Kapor

“Arguing that you don’t care about the right to privacy because you have nothing to hide is no different than saying you don’t care about free speech because you have nothing to say.”
–Edward Snowden

Abstract

Today, many are unaware of how much of their personal information is publicly available on the web, which has become an increasingly important issue among internet users. This thesis builds on the work of the preceding Capstone project and uses the open-source Online Privacy Pilot tool as a case study to explore how large language models can be incorporated into the tool to enhance its functionality and assist users in managing their online fingerprint.

Based on our evaluation of the Mistral and Llama 2 models, we selected Mistral and incorporated it into three features of the Online Privacy Pilot tool: generating recommended positive filters, clustering user profile entries, and creating informative snippets for these entries. The recommended positive filters are generated based on the entries in the user profile and allow the user to provide relevance feedback to the tool if they choose to add them to the search query. Additionally, we selected and proposed a total of 13 cluster labels for use in the tool's clustering feature. To address ethical and legal considerations, especially concerning user intent and data privacy, we implemented an additional step when adding footprints to the user profile, guiding users to store only personally relevant footprints.

Acknowledgements

I would like to extend my sincerest gratitude to my supervisor, Elisavet Kozyri, whose guidance and insightful feedback have been invaluable throughout this process. I would also like to thank my classmates and friends, who have made the past five years enjoyable and memorable. Sharing this journey with you has truly been a highlight. To my family, thank you for your support and encouragement over the years. Your belief in me has been a source of motivation.

Thank you all for being part of this significant chapter of my life.

Contents

List of Figures

List of Tables

1 Introduction

In today's digital age, where the internet is used for many purposes by a large number of users, many are unaware of how much personal information about them is publicly available on the web. With this in mind, it has become important to help individuals get better control over their online presence by providing them with an overview of their public information.

To address this issue, Benoît Leconte and Daniel Nicolas Pressensé developed an open-source tool called the Online Privacy Pilot (OPP) tool [2] during their internship at the Department of Computer Science at UiT in 2023. The OPP tool was used as a case study for the Capstone project that precedes this Master's thesis, and the results and source code from that project are used as a basis for this thesis.

The OPP tool can be divided into two main components: the explorer which is in charge of locating the user's footprints based on the search query, and the presenter which is in charge of presenting the results to the user.

The presenter also allows the user to enter a search target, for example their name, and a set of other search parameters. Based on the provided search parameters, the explorer crawls the web for digital footprints related to the search query. The results are afterward displayed to the user in a graph structure, where each node in the graph corresponds to a potentially relevant digital footprint for the user.

These digital footprints are traces that the user leaves behind online. This could for example be a social media post or some other public information. A unique collection of the digital footprints left behind by the user can then be seen as their online fingerprint.

From a footprint located by the OPP tool, the user has the opportunity to be guided to the source of the corresponding public information to delete it or change the related privacy settings. This feature is an important part of the OPP tool as it helps the user gain better control over their online presence by removing information that they do not want to have publicly available on the web.

The focus of the Capstone project that precedes this Master's thesis was on incorporating new features into the OPP tool to further improve the user experience and relevance feedback given by the user. The features that were implemented were based on ideas from information retrieval literature. These features include a user profile to which the user can choose to add relevant results. With the user profile, other features were also implemented to give the user a better overview of their online fingerprint. These include clustering of profile entries, snippet generation for profile entries, the ability to check if a profile entry is still alive, and recommended positive filters that are generated based on the entries stored in the user profile.

1.1 Problem Statement

This thesis aims to explore how semantic search techniques can be incorporated into the OPP tool with the use of large language models (LLMs) to further improve the retrieval and presentation of a user's online fingerprint.

The main research questions for this thesis are:

A. How can LLMs be used in the incorporation of semantic search techniques to improve the OPP tool's retrieval and presentation of a user's online fingerprint?

B. How does the storage of the user profile impact user privacy, and how can we guide the user to not store entries related to someone else?

C. What cluster labels, that are more fitting for the OPP tool's purpose, can be used when clustering the user profile entries?

Research question A. can be further divided into the following questions:

A1. How can LLMs be used to understand user intent and perform semantic query expansions that include synonyms and contextually related terms in the search query, to decrease the number of irrelevant results?

A1.1. How can LLMs be used to improve the process of generating recommended positive filters based on the entries in the user profile?

A2. How can LLMs be used to automate the clustering of a user's profile entries?

A3. How can LLMs be used to generate snippets of profile entries that are more useful to the user?

1.2 Contribution

The contributions of this thesis involve further improving the OPP tool's process of locating and presenting a user with their public information from the web by exploring literature from the semantic search and LLM research areas.

Inspired by the literature, the following changes are made to the OPP tool:

- To address research question A1., and specifically subquestion A1.1., the LLM Mistral is utilized through the Ollama API to increase the OPP tool's semantic understanding of user queries by suggesting semantically related keywords that the user can add to the search query. This implementation includes the automation of the generation of recommended positive filters, which are based on the user profile entries.

- To address research question A2., Mistral is used to automate the clustering of user profile entries. This implementation is motivated by the need for better organization of the user profile, which makes it easier for the user to examine and manage their online fingerprint.

- To address research question A3., the snippet generation feature of the OPP tool is refined through the use of Mistral. This change aims to provide the user with more concise and informative snippets of their user profile entries, allowing for a better overview of the content of the entries.

- To address research question B., a privacy-preserving step is introduced to the OPP tool. This step involves the implementation of additional confirmation before adding a search result to the user profile so as not to accidentally store entries related to someone other than the user of

the OPP tool. This step is also added to minimize the risk of mistakenly storing information related to other individuals and to guide the user to not deliberately store entries related to someone else.

- To address research question C., 13 cluster labels are selected for the OPP tool's clustering feature, of which eight are inspired by existing literature related to online user activities, and the remaining five are proposed by us. These clusters are selected to provide the user with an organized view of their online fingerprint.

1.3 Context

This Master's thesis is written as part of the Cyber Security Group (CSG) at UiT - The Arctic University of Norway. CSG divides the group's research into three main areas: fundamental systems, system support for healthy human beings, and system support for sustainability [4]. The group's work also focuses on mobility, social networking, multimedia, cloud computing, and artificial intelligence.

The Online Privacy Pilot tool is used as a case study for this thesis. The tool was developed by Benoît Leconte and Daniel Nicolas Pressensé during their internship with CSG at the Department of Computer Science at UiT in 2023.

This Master's thesis builds on the preceding Capstone project [1] from the fall semester of 2023 titled *Information Retrieval Techniques for Managing Online Fingerprint*.

1.4 Thesis Outline

The rest of the thesis is structured as follows: Chapter 2 provides background information related to the thesis and related work. Chapter 3 focuses on the case study for this thesis, the Online Privacy Pilot tool, and explains the tool's functionality. Chapter 4 gives an overview of the work done during the Capstone project that precedes this Master's thesis and Chapter 5 presents the design and implementation of this thesis. Chapter 6 evaluates the changes made during this thesis, and Chapter 7 discusses topics related to the thesis and future work. Finally, chapter 8 concludes this thesis.

2

Background and Related Work

This chapter provides an overview of background information and work related to this thesis and the functionality of the OPP tool.

2.1 Information Retrieval

Information retrieval is an important research area for this thesis as it forms the foundation of the OPP tool's main purpose of retrieving and presenting public information related to the user. The field of information retrieval involves finding and retrieving documents that satisfy a user's information need [5, p. 1]. This section explores several techniques from the field of information retrieval that are used during the OPP tool's process of retrieving and displaying relevant footprints to the user, such as semantic search, web crawling and scraping, relevance feedback, clustering, and snippets.

2.1.1 Semantic search

Semantic search techniques aim to improve the accuracy of search results by gaining a better understanding of a given query in regard to user intent and

the contextual meaning of search terms. As traditional information retrieval systems would base the retrieval of documents on the occurrence of terms [27], they would often retrieve results irrelevant to the search query. Semantic search techniques can mitigate this issue by taking advantage of the meaning of search terms or their relations [28], and in this way retrieve documents that are more likely to fit the user's information needs.

Semantic search techniques are important in tools such as the OPP tool, allowing for a better understanding of the meaning of terms and user intent, for example, by finding terms related to the entries stored in the user profile.

2.1.2 Web crawling and scraping

The OPP tool employs web crawling and scraping to find and extract information related to the user based on provided search parameters.

Web crawling and web scraping are often used interchangeably as they are two closely related concepts [29, p. 155]. They are however two distinct processes with different use cases.

Web scraping focuses on extracting content from individual web pages and involves the automatic retrieval, parsing, and organizing of data [29, p. 3]. Using a computer program to perform web scraping is more efficient than having to manually open a web page and copying its contents. While many websites offer access to structured data through an API, they may not always be available or may not provide access to the needed data, making web scraping a more suitable tool [29, p. 4-5]. Web scraping can be used for various applications, for example, to gather data for analysis by researchers or for commercial applications to keep track of store prices [30].

While web scrapers focus on a single web page, web crawlers broaden this scope by navigating multiple pages, ranging from a set of web pages belonging to a single website to more open-ended crawling where the crawler is not limited to web pages under a single domain [29, p. 155-156]. Unlike web scrapers, where the goal is to extract data from a web page, web crawlers do not typically have a specific, well-defined goal, but rather focus on traversing the web while locating a range of potentially relevant web pages [5, p. 443], and create a map of the traversed web pages that can later be scraped. Web crawlers typically start with a set of starter URLs and find new URLs to explore from the contents of those URLs [5, p. 444-445].

2.1.3 Relevance feedback

Allowing the user to provide relevance feedback to the OPP tool is important to refine its search process when retrieving potential relevant footprints.

A central part of information retrieval systems is finding information that is relevant to a given information need. However, it can be difficult for these systems to determine what piece of information is relevant without sufficient context. Relevance judgments may also vary from user to user as they are subjective [5, p. 167], and might even change as the user looks through the results returned by the system. Users may also make faulty judgments of relevance if they do not have the necessary context or if they misunderstand the contents of a result.

To aid the system in finding results that are relevant for an information need, the user may provide feedback on the relevance of results, for example by marking them as either relevant or not relevant [5, p. 178]. By giving relevance feedback, information retrieval systems may also be better equipped to follow the user's information needs and ideas of relevance as they change. Another way for users to give relevance feedback could for example be by including additional search terms in the query that are recommended by the information retrieval system [5, p. 189].

2.1.4 Clustering

Clustering is helpful in the OPP tool as it allows for a more organized view of the user profile, giving the user a better overview of the entries.

In information retrieval systems, clustering involves grouping semantically similar documents together [31, 32], for example by a common property, and can be used when presenting search results to the user or during the search process itself. This is a helpful feature as it allows for a more effective and organized view of results, making them easier for the user to look through [5, p. 350-351], especially in cases where the system returns a long list of results.

2.1.5 Snippets

By finding and displaying snippets of the entries in the user profile, the OPP tool can give the user a better context and understanding of the stored footprints, which is helpful when giving relevance feedback back to the system.

In information retrieval systems, snippets are short summaries of documents created to assist users in determining the relevance of a document and making a list of results more informative [5, p. 170]. Snippets are especially helpful as they can give the user a better understanding of a document without having to manually open it and look through its content. A challenge when creating snippets in information retrieval systems is therefore including information that is helpful to the user and provide enough context for the user to understand what the document is.

Snippets can be divided into two main types: static and dynamic snippets [5, p. 171-172]. Static snippets are fixed summaries of documents, with the same content regardless of the user's query, typically including a few sections of the documents, such as the title or the first sentences. The content of dynamic snippets on the other hand, includes information that may be relevant to the user's query, for example, extracts of the document that contains terms used in the query.

2.2 Large Language Models

Large language models (LLMs) have quickly gained popularity in many areas, including personal, academic, and commercial applications, and are now available with a wide variety of models. These models are trained on large sets of data and can complete tasks such as text generation, question answering, and information summarizing [26].

LLMs are typically characterized by their number of parameters, which represent information learned from the training data [49]. These parameters, which are often in the billions, influence a model's ability to understand and generate text.

LLMs, specifically the models Llama 2 and Mistral, have been selected for this thesis. These models were chosen because they are both open-source, allowing for running them locally through platforms such as Ollama [38] and for potential fine-tuning in the future. Open-source models also provide transparency, which is important when considering the privacy and ethical aspects of the OPP tool. Additionally, both models are relatively new, incorporating some of the more recent research in the field of LLMs.

Chang et al. [26] discuss evaluating LLMs by focusing on what, how, and where to evaluate them, based on existing work on LLM evaluation. The authors found that LLMs displayed some limitations, especially in terms of reasoning and robustness. They saw that the LLMs were sensitive to variations in given

prompts and that they may use fabricated information in their responses [26]. However, with the rapid development in the field of LLMs, these limitations may soon be overcome, while new challenges could emerge.

2.2.1 LLM hallucinations

An LLM hallucination is a behavior that may be exhibited by an LLM where it generates false or inaccurate statements [26, 34]. While these hallucinations may look correct at first glance, they contain false information generated by the model, making it a large challenge when incorporating LLMs into applications that require reliable information [35, 36, 37], such as information retrieval systems.

Zhang et al. categorize hallucinations into input-conflicting, context-conflicting, and fact-conflicting hallucinations [37]. They define input-conflicting hallucinations as responses that are irrelevant to the given input, context-conflicting hallucinations as responses that introduce information not previously mentioned, and fact-conflicting hallucinations as responses containing false information. Of these, fact-conflicting hallucinations have received more focus in research [37]. To mitigate these issues, strategies such as refining training data and employing prompt engineering to guide the models' responses have been explored. For example, Touvron et al. [39] reduced hallucinations by explicitly instructing the LLMs not to generate false information in their prompts.

Addressing the issue of hallucinations or false information generated by LLMs is important for maintaining trustworthy and factual results when incorporating the use of LLMs into the OPP tool.

2.2.2 Prompt engineering

Prompt engineering plays a crucial role in the use of LLMs, especially because models can be sensitive to variations in the prompts they receive [26, 37]. It involves designing and writing prompts with specific instructions to guide the LLMs toward the wanted responses or to perform required tasks. Having become an important area of research itself, prompt engineering explores various methods and techniques [46, 47] to improve LLMs' responses.

Chen et al. [46] present an overview of some methods used during prompt engineering to increase the efficiency of LLMs. These methods range from writing prompts clearly and precisely to more advanced methods such as "Chain of Thought" prompting [48] where the LLM includes a set of reasoning steps when generating its response.

As the prompts sent to the LLMs have a big effect on the generated responses, it is an important consideration when incorporating LLMs into the OPP tool to ensure that the responses are relevant and reliable.

2.2.3 Llama

Llama 2, an open-source LLM, was developed and released by Meta in 2023. Based on the release publication for the model [39], Llama 2 displays a significant update from the previous model, Llama 1, with improvements such as a larger training dataset and an increased context length. The increase in context length allows Llama 2 to process more information than the previous model, giving it more context when generating its responses. With its improvements, Llama 2 can perform a range of tasks with more complex and contextually relevant responses. Meta released several versions of Llama 2, ranging from 7B to 70B parameters.

Llama 3 is Meta's newest open-source model, released in April 2024 [42]. The model shows several improvements from Llama 2 and has been trained on a dataset seven times larger than its predecessor. Llama 3 was released with model sizes of 8B and 70B parameters.

2.2.4 Mistral

Mistral 7B, developed by Mistral AI and released in 2023, is an open-source model with seven billion parameters. According to benchmarking performed by Jiang et al. [43], Mistral 7B outperformed the Llama 2 13B model, especially in terms of efficiency and performance. This efficiency makes it a good option for applications where computational resources are a concern. However, its relatively limited parameter count may also restrict its capacity to store information from pre-training compared to the larger models.

2.2.5 ChatGPT

ChatGPT, developed and released by OpenAI [44], is a model specifically fine-tuned to generate human-like text in conversational contexts. The model is built on the GPT-3 model [45], also developed by OpenAI, which is known for its broad training on a diverse set of internet texts. With this training, ChatGPT can effectively perform a wide range of conversational tasks. The model has capabilities such as recognizing errors, asking follow-up questions, and refusing improper requests [44]. These abilities make ChatGPT effective for applications where conversational abilities are the focus.

2.2.6 Ollama

Ollama is an open-source project [40] that offers an easy-to-use platform for running LLMs locally and is available on Windows, macOS, and Linux [38], making it a good option for running LLMs locally when incorporating them into the OPP tool. Ollama supports a varied list of models [41], including Llama 2, Mistral, the newly released model Llama 3, and many more that can be accessed through Ollama's REST API programmatically.

2.2.7 LLMs for information retrieval

LLMs can be useful for tools such as the OPP tool because they can better understand the semantics of terms, for instance by including semantically related terms during the process of generating recommended positive filters. They can also be helpful when performing other information retrieval tasks.

Previous work has been done in the area of incorporating LLMs into information retrieval systems, either during the search process itself or for other more specific tasks. WebGPT [34], for example, was designed as a way to use an LLM for performing the search in information retrieval systems. It uses a fine-tuned GPT-3 model to answer a user's questions, by sending queries to the Microsoft Bing Web Search API [34] to retrieve documents for its answer. LLMs have also been used to perform other information retrieval tasks. For example, Wang et al. presented Query2doc [35], a way to use LLMs for performing query expansions in information retrieval systems by adding pseudo-documents generated by LLMs to the original query.

There is also previous work on improving LLMs' ability to perform information retrieval tasks. Zhu et al. proposed INTERS ("INstruction Tuning datasEt foR Search") [33], a dataset used to fine-tune LLMs and improve their performance in information retrieval systems. The INTERS dataset was created with a focus on understanding the query, documents, and the relationship between the two, which are three important aspects of information retrieval systems.

Where these examples of previous work focus on the usage and effectiveness of LLMs in different areas of information retrieval systems, the OPP tool incorporates LLMs into its process of finding and presenting a user's digital footprints.

2.3 Web Page Categorization

Previous research on web page categorizations could be relevant to the clustering feature of the OPP tool. We were unable to find proposals for standard category labels to use during the clustering, but the following work has selected or proposed categories that fit their specific use cases.

Chaker and Habib [52] explored how web pages could be categorized by assigning them to all predefined categories but with different weights for each category. They determine a web page's genre using two classifiers: contextual, which uses the web page's URL, and structural, which uses the structure of the web page. For the predefined categories, they used two datasets: KI-04 and WebKB. From the KI-04 dataset, the authors based their categories on eight categories that were determined through a user study on genre usefulness by Meyer zu Eissen and Stein: "help, article, discussion, shop, portrayals of companies and institutions, private portrayal, link collection, and download" [53]. From the WebKB dataset, which contains web pages from computer science department sites from American universities [54], Chaker and Habib used six categories: project, student, staff, course, department, and faculty, excluding the category other.

Although these categories may be useful for a limited range of web page types, they do not cover all web pages encountered when using the OPP tool, given the wide range of web page types a user's footprints may be located at.

Some studies on URL classifications have used web directories such as the Open Directory Project, which is a directory of web pages that have been classified by humans, but it has since been shut down. After the closure of the Open Directory Project, a successor project was made public [56] which includes the 15 main topics from the original directory.

For instance, Baykan et al. [55] looked at the problem of identifying a web page's topic based only on its URL, and evaluated different techniques and algorithms using five datasets, including the Open Directory Project. From the Open Directory Project, they used web pages from the directory's 15 main categories during their evaluations.

While using categories from web directories such as the Open Directory Project would cover a wider range of web page types, some of these categories might be outdated and might not include, for example, the social media category.

2.3.1 Online activities

Given that earlier web page categorization proposals were not applicable to the OPP tool's use case, the focus shifted to researching what types of activities people engage in when online. When looking for such activities, several internet surveys carried out in the United States [9] and the United Kingdom [7], as well as other international surveys [8] where several countries participated, were found. These surveys are carried out as a collaboration under the World Internet Project [10], which was founded in 1999 by the USC Annenberg School Center for the Digital Future in the United States. The World Internet Project focuses on the impact the internet has on social, political, and economic areas [10] in various participating countries around the world.

Blank and Groselj [6] looked at the amount, variety, and types of internet use in Britain based on a dataset published by the Oxford Internet Surveys (OxIS) [23] in 2011 and the Oxford report by Dutton and Blank [24] from the same year. OxIS carries out the United Kingdom's contribution to the World Internet Project. In their paper, Blank and Groselj look at a set of 48 variables for internet activities from the 2011 dataset. They then identify the following 10 types of internet activities based on this set: entertainment, commerce, information seeking, socializing, email, blog, production, classic mass media, school-work, and vice [6]. Each of these types consists of multiple activities such as social networking, reading and writing blogs, looking for news and sports information, and looking for information for school and work.

In 2019, OxIS published a survey on threats to privacy online [7] based on internet use in Britain. With a focus on privacy, participants were mostly asked about their commercial, entertainment, and content production activities when online. This includes activities such as watching movies, posting videos, commenting and posting content on social media, blog writing, and maintaining websites. Participants were also asked about what platforms they carried out these activities on.

The United States is one of the major contributors to the World Internet Project through the Center for the Digital Future, where they have published multiple reports, both Digital Future Project reports based on data from the United States and World Internet Project reports based on data from numerous cooperating international partners [25]. The Center for the Digital Future published its 16th annual study [9] in 2018 on the impact the internet and other technology have on Americans, based on data from participants in the United States. Participants were asked about activities from five areas: "Social networking", "Fact-finding, information sources, and education", "Information gathering", "eCommerce", and "Entertainment and personal interests" [9]. These areas include activities such as posting on discussion boards, posting and interacting with content on

social media, getting information for school or work, looking for news, reading blogs, watching videos, and playing games. The participants were also asked about other aspects of internet use, such as the amount, and when and where they accessed it.

In 2018, the ninth edition of the World Internet Project Report was published by the Center for the Digital Future [8]. This is an international survey based on data from multiple collaborating countries. In regards to activities on the internet, participants were asked about activities from the areas of communication, research, school-work and distance learning, buying and selling, financial management, entertainment, and personal interest. These areas include activities such as posting content, looking for news, selling items, watching videos, and playing games [8].

From these surveys and reports, it was seen that various types of activities, including work, education, news, sports, commercial, social media, entertainment, and blogs were often used to cover activities that online users might participate in.

2.4 Open-Source Intelligence

Open-Source Intelligence (OSINT) techniques, employed by the OPP tool in its search process, is an important topic for this thesis. The use of OSINT techniques highlights the importance of limiting the use of tools such as the OPP tool to non-malicious purposes.

OSINT is the method of collecting and analyzing information from public sources such as social networks, public websites, papers and public publications, as well as other publicly available platforms [22]. Wikipedia is an example of an OSINT collection, as it stores publicly available information that anyone can access [21].

As OSINT techniques involve the use of public sources, they can be used by anyone. For example, law enforcement and governments use these techniques to detect and fight various forms of cybercrime and other criminal activities [21, 22].

While OSINT techniques can be used for positive purposes, they can also be exploited and used for malicious purposes such as various forms of cybercrimes [21]. They can for example be used by an attacker to gain information on a target during social engineering attacks [22]. It is therefore important to limit the use of OSINT techniques to non-malicious purposes, for example when

using tools such as the OPP tool, and to consider the legal and privacy aspects of OSINT techniques.

2.5 Online Privacy

Online privacy is crucial in the use of the OPP tool, which aims to provide users with a better overview of their online presence and their publicly accessible information.

In today's digital age, online privacy, which can be seen as a fundamental human right, faces many challenges [20]. With the large amount of personal information that is made publicly available on the web every day, many users lose control over their information.

The ethical and legal considerations surrounding OSINT, as previously mentioned, emphasize the sensitivity of this publicly available information. Although this information is publicly available, it does not mean that it is not sensitive information [22], further emphasizing the need to limit the use of OSINT to non-malicious purposes. It also highlights the need to handle such data carefully, by following laws and respecting data protection policies such as the EU General Data Protection Regulation (GDPR) when using and developing tools that employ OSINT techniques.

Some online archives have been created to store the "history" of the web by taking snapshots of web pages. The Wayback Machine is one such archive that was started by the Internet Project [57]. It has archived a large number of web pages, including multiple versions of the same pages that have existed over the years. While the archive only stores public web pages, it may still include personal information that a user may no longer want to be available. In such cases, the user can send a request to the Internet Project to have the information removed from their archive. In addition, to address privacy considerations, the Internet Archive's terms of use, privacy policy, and copyright policy [58] include points such as agreeing to not collect or store personal data about others, and to not violate other's rights of privacy.

2.5.1 Retrieving and removing personal information from the web

The OPP tool aids users in finding and removing their public personal information. Other tools and services have also been created to retrieve or remove personal information from the web or from data brokers, which are companies

that collect and analyze personal data from multiple sources and then sell the insights they gain from this data [19].

The SINCE engine [18], for example, is a crawler tool that collects data from the web. However, where the OPP tool retrieves data from various sources on the web, the SINCE engine only collects data from public Facebook pages and looks at interactions between users and content. SINCE and OPP are therefore both crawler tools, but they have different use cases and goals.

Services known as Personal Identifiable Information (PII) scrubbers [15, 16], aim to help users remove personal data and regain control over their online privacy. As the name suggests, these services help users scrub personally identifiable information, which is information that can be used to identify an individual. Where the OPP tool aims to give the user an overview of where their information is located in the form of footprints, and where possible, guide them to where they can remove that information themselves, these PII scrubbers mainly focus on the removal of personal data from data brokers [15, 16].

Incogni [15] and DeleteMe [16] are two examples of PII scrubbers that send removal and opt-out requests to a list of data brokers. Both services require a paid subscription to use. DeleteMe also offers a "Search Yourself" service [17] that performs a Google search for a user-provided name. The user is then shown a list of search results and can choose which results they want DeleteMe to remove. This service is similar to the OPP tool in that it also uses Google to locate information that may be relevant to the user. Customers do however have to provide personal information to use Incogni and DeleteMe, and in some cases, this information might be included in the requests that are sent to the data brokers, which are important considerations before using such services.

2.5.2 JustDeleteMe

The OPP tool includes a "Delete me" feature that guides users through the account deletion process for web services listed in the JustDeleteMe directory [11]. This directory is a part of the JustDeleteMe project, an open-source project started by the JustDeleteMe Contribution Team, that maintains a directory of web services and information on how to delete user accounts from those services. The directory may, for example, provide a link to the specific web page where the user can delete their account after logging in, or by providing other forms of identification depending on what the web service requires. The JustDeleteMe project also offers a website [12] where the user can see an overview of the web services in the JustDeleteMe directory, with entries color-coded based on how difficult the deletion process is, ranging from green

(easy) to black (impossible). Where possible, the overview also includes a link to where the user can delete their account along with additional information about the deletion process.

The JustDeleteMe Team also maintains two similar directories called JustGetMyData [13] and JustWhatsTheData [14]. The JustGetMyData directory provides information on how to obtain your data from web services, while the JustWhatsTheData provides information about the amount and type of data different web services collect from you. Since these three projects are open-source, users can choose to contribute by adding new web services to the directories or making changes to existing ones.

3

Online Privacy Pilot

This chapter presents the open-source [2] Online Privacy Pilot (OPP) tool and its original features. User documentation for the OPP tool can be found on GitHub [3].

3.1 Overview

The OPP tool was developed by Benoît Leconte and Daniel Nicolas Pressensé during their internship with CSG at the Department of Computer Science at UiT in 2023. The tool collects public information that is potentially relevant to the user based on provided search input.

The tool can be divided into two main components: the explorer and the presenter. The explorer's main task is to crawl and scrape the web for information that is potentially relevant to the user based on the provided search parameters. The presenter's main task is retrieving the collected information from the explorer and displaying it to the user.

The backend of the OPP tool is implemented in Python and the frontend is developed using the React framework. Interactions between the backend and frontend (i.e., explorer and presenter) are performed through the backend's Rest API.

3.2 The Explorer

Based on the user-provided search parameters it receives from the presenter, the explorer performs web searches looking for potentially relevant information. The search parameters consists of a main target, positive and negative filters, whether the user wants the OPP tool to perform an active search or not, and the wanted search depth. The user inputs the wanted search parameters in the tool's search form as seen in Figure 3.1.

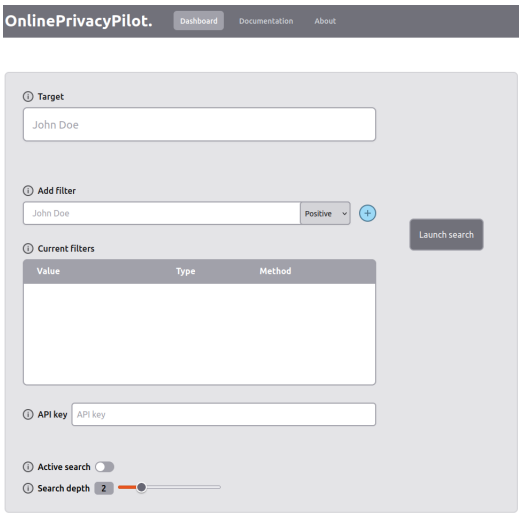

Figure 3.1: Original OPP search form.

The main target of the search can for example be the user's name. If any positive filters are added to the search parameters, the explorer will only retrieve results that contain those filters. If any negative filters are added, the explorer will exclude results that contain those filters. This means that if the user adds a positive filter of *UiT*, the explorer will only retrieve results that contain the term *UiT*. If the user for example adds *LinkedIn* as a negative filter, the explorer will exclude results that contain the term *LinkedIn*. Figure 3.2 shows the search form of the OPP tool with Emmanuel Macron as the search target, "twitter" as a negative filter, and "instagram" as a positive filter. Where needed for examples in this thesis, Emmanuel Macron is used as the search target because he is a well-known, public figure.

The active search option allows the user to choose whether the OPP tool should use OSINT techniques during the search process or not, while the search depth specifies the number of recursions that the explorer will perform on the provided search target during the search process.

To use the OPP tool, the user has to supply their own Google API key, which is

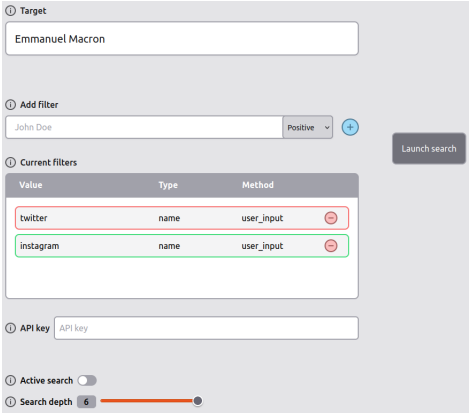

Figure 3.2: Search form with positive and negative filters.

used by the explorer to authenticate against the Google API during the search process. With this requirement, all requests that the user makes through the OPP tool will be linked to their Google account.

3.3 The Presenter

The presenter is in charge of displaying the results found by the explorer. These results are displayed as a graph structure where each node corresponds to a potentially relevant digital footprint for the user. An example graph found by using Emmanuel Macron as the search target is displayed in Figure 3.3. The OPP tool is not able to store results from multiple searches but will mark newly added or moved nodes in the result graph in a red color between two searches. Figure 3.4 shows the result graph after adding "news" as a positive filter to the search query. New footprints found with this search are marked as red nodes in the graph.

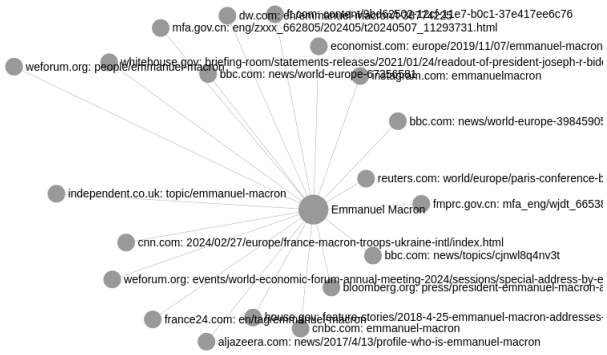

Figure 3.3: Search results with Emmanuel Macron as the search target.

Figure 3.4: Search results with Emmanuel Macron as the search target and "news" as a positive filter.

The user can click on the nodes of the graph to mark the corresponding footprint as relevant or irrelevant, adding it as a positive or negative filter respectively. This feedback is then sent to the explorer to help it refine its search process. By clicking on a node in the result graph, the user can also see additional information about the corresponding footprint, such as what type of footprint it is and what method was used to find it. Figure 3.5 displays the information modal for Emmanuel Macron's Instagram page.

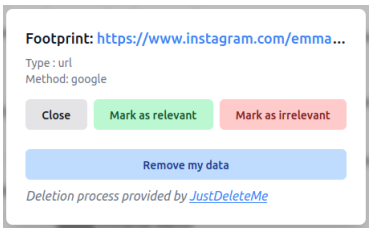

Figure 3.5: Information modal for Emmanuel Macron's Instagram page.

The presenter is also responsible for guiding the user to the source of the information where possible. This feature can be accessed by clicking on a node in the result graph and clicking the *Remove my data* button, as displayed in Figure 3.5. This button is available if the clicked entry is recognized as a URL or is linked to a user account on a website, and also found in the JustDeleteMe database [11]. This functionality allows the user to decide which parts of their online fingerprint they want removed, or if the respective privacy settings should be changed to make the information non-public.

4

Capstone Project

This chapter presents the changes that were made to the OPP tool during the Capstone project from the fall semester of 2023 [1] which precedes this Master's thesis. The following implementation acts as a basis for this Master's thesis.

4.1 User Profile

The biggest part of the Capstone project was the introduction of a user profile. The user profile allows the user to store relevant footprints and get an overview of their online fingerprint.

The user can add a footprint to the user profile from the results graph. This is done by clicking on a node in the graph, which opens up an information modal with a button for adding it to the profile. Figure 4.1 shows the information modal for Emmanuel Macron's X (formerly Twitter) page. If a footprint has already been added to the user profile, the add to profile button is changed to a button for removing the footprint from the profile. The user profile can be accessed through the navigation bar of the OPP tool, where a list of the stored entries is displayed, as seen in Figure 4.2.

The database containing the user profile entries is stored locally and managed with SQLite. The user profile is stored between runs, but the user can choose to

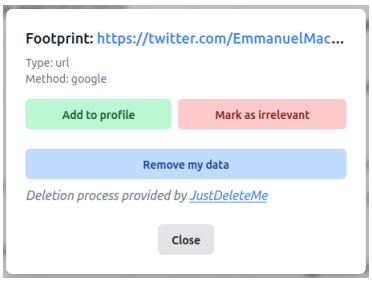

Figure 4.1: Information modal for Emmanuel Macron's X (Twitter) page.

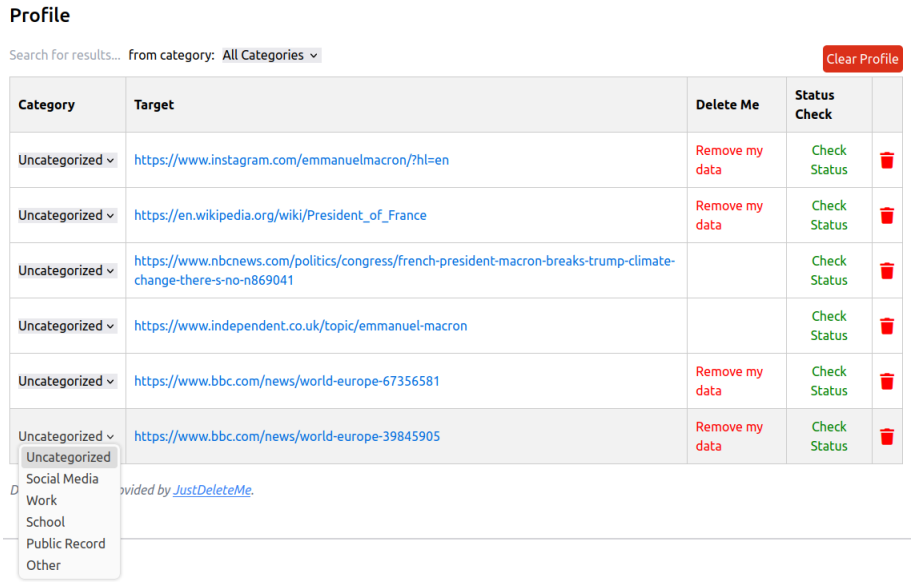

Figure 4.2: Populated user profile with Capstone project implementation.

delete all entries in the database through a clear button, or by deleting entries individually through the user profile page. The user is only given a single user profile and is not able to create multiple profiles as the user is only supposed to store footprints related to themselves.

For entries that are recognized as a URL or connected to a user account, in addition to being found in the JustDeleteMe database, the user can be guided through the deletion of the public information corresponding to the profile entry. In cases where these terms are fulfilled, a button with the text "Remove my data" is displayed on the respective profile entry's row in the list, as seen for some of the entries in Figure 4.2.

4.2 Recommended Positive Filters

The generation of recommended positive filters was also implemented in addition to the user profile. This implementation generates keywords that are displayed as a list of recommended positive filters to the user in the search form. This list was added to the right side in the OPP tool's search form, as displayed in Figure 4.3. As seen in the figure, a message is displayed if no recommended positive filters are available, for example, if there are no entries in the user profile. This is because the recommended positive filters are generated based on the entries stored in the user profile and are created by parsing and extracting keywords from the entries' URLs. Giving the user this list of potentially relevant filters allows for an easier way of providing relevance feedback to the OPP tool. Figure 4.4 shows four of the recommended positive filters generated for the profile entries in Figure 4.2. With this feature added to the OPP tool, the original button for marking a result in the graph as relevant was removed as the user can choose to add positive filters through the generated list of recommended positive filters instead.

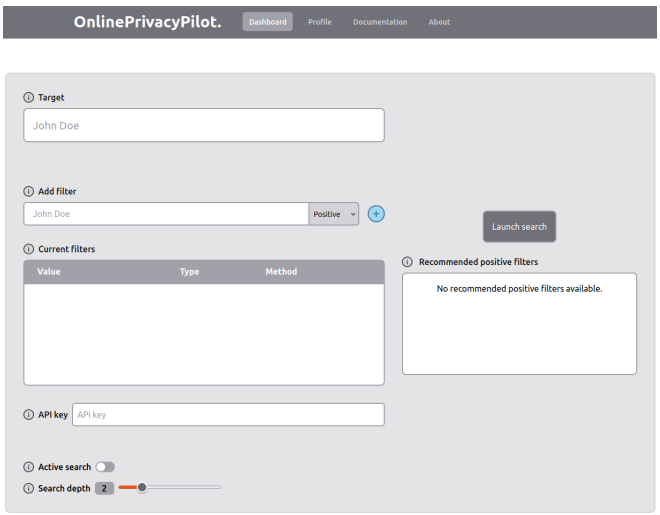

Figure 4.3: The updated version of the OPP tool's search form for the Capstone project.

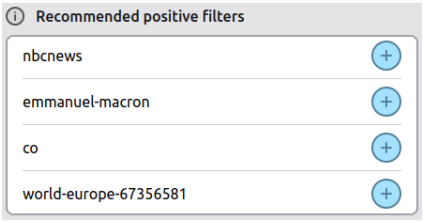

Figure 4.4: Four examples of the recommended positive filters generated by the OPP tool for the profile entries in Figure 4.2.

4.3 Clustering

A clustering feature was implemented for the user profile, where the user can manually assign clusters to the profile entries. The menu for manually selecting a cluster is displayed on the bottom left side of Figure 4.2. When the user adds a footprint to the user profile, the new entry is assigned to an "Uncategorized" cluster. The user can then choose to assign the entry a new cluster from the following: Social Media, Work, School, Public Record, and Other. The profile entries can be clustered based on these assignments through either filtering or sorting.

4.4 Snippets

A snippet feature was also implemented for the user profile. This feature allows the user to click on an entry in the profile to see a snippet of the web page corresponding to that entry.

When the user clicks on an entry to see its snippet, a request is sent to the backend. The backend then makes sure that the entry contains a URL, and if not, lets the user know that no snippet could be retrieved. If it has a URL however, the backend retrieves a part of the web page's content to be displayed to the user. If the URL belongs to a social media website such as Instagram, TikTok, or LinkedIn, the backend uses the respective scraper that was created for the original version of the OPP tool. For URLs that do not belong to these social media pages, the HTML content of the web page is retrieved, and the Python library BeautifulSoup is used to find the title and first paragraph of the page. The snippet is finally returned to the frontend and displayed to the user in a modal along with the corresponding URL. Figure 4.5 shows an example snippet that is generated for a news article related to Emmanuel Macron from NBC News.

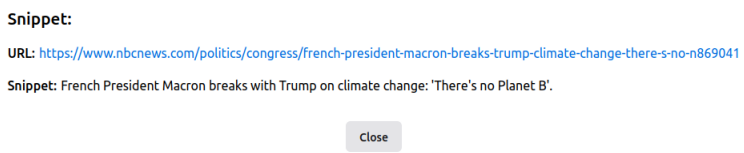

Figure 4.5: Example of a snippet generated for a news article from NBC News that is related to Emmanuel Macron.

4.5 Status Check

A status check feature was also added to the user profile during the Capstone project. This feature allows the user to check if information corresponding to a profile entry is still publicly available on the web by clicking the "Check Status" button on an entry in the user profile, as seen on the right side of the populated user profile in Figure 4.2. When clicking this button, a request is sent to the backend to check the entry's current status.

The process of checking if the information belonging to an entry in the user profile is still publicly available is done by sending a request to the respective web page and checking the returned status code. With this process, many edge cases have to be considered, such as websites employing different status codes for the same purposes, or the tool encountering a CAPTCHA test. To handle such cases, the OPP tool was made to be restrictive in that it is more likely to return false positives (i.e. saying that the information is still publicly available when it is not) instead of false negatives (i.e. saying the information is no longer publicly available when it is), as a false negative will lead the user to wrongly believe that their personal information is no longer publicly available when it is.

The determined status of the profile entry is finally returned to the frontend, and is displayed to the user in a modal along with the entry's URL, as seen in Figure 4.6. The figure shows that the OPP tool determined that Emmanuel Macron's Instagram page is still publicly available.

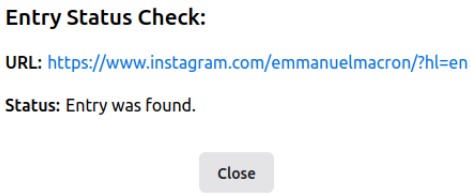

Figure 4.6: Status check modal for Emmanuel Macron's Instagram page.

5

Design and Implementation

This chapter presents the changes made to the OPP tool for the Master's thesis. The following implementations are integrated into the OPP tool, replacing or improving the previous implementations from the Capstone project.

5.1 Storing Profile Entries

For this thesis, the focus when working on the storage of user profile entries is on privacy. This work focuses on incorporating privacy-preserving steps into the OPP tool, by considering the ethical and legal aspects of storing personal information, in this case in the form of a user's digital footprints.

In addition, a change is made regarding what entries can be stored in the user profile. For the Capstone implementation, all nodes in the results graph could be added to the user profile, which includes both URL and non-URL nodes. A change is however now made so the user can no longer add non-URL nodes to the user profile, only nodes with a URL. This change is made because we want to provide the user with an overview of their digital footprints corresponding to information that is located on public web pages. Nodes that do not have a URL are therefore not included as they are offered only as additional information

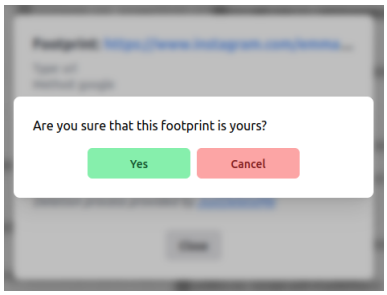

Figure 5.1: Confirmation modal when adding a footprint to the user profile.

for the actual footprints.

The main focus for this thesis' code implementation regarding the storage of profile entries involves the introduction of an additional step when adding a footprint to the user profile. This step is implemented in the form of a double confirmation modal as shown in Figure 5.1.

This extra step is helpful when considering that a user could store footprints related to someone other than themselves, and is added to guide the user to only store footprints about themselves.

5.2 Recommended Positive Filters

Based on the entries in the user profile, a set of recommended positive filters are generated and displayed to the user in the search form of the OPP tool, as seen in Figure 5.2. These recommended positive filters are generated for each profile entry when added to the user profile, and all the entries' filters are combined into a unified list when requested by the frontend to be displayed to the user. Previously, the recommended positive filters were all generated when the search form was loaded or when changes were made to the user profile, which gave a large time overhead when using the tool. By generating and storing the filters individually for each entry in the user profile, the OPP tool can track which filters belong to which entry and avoid having to regenerate all filters when the user profile is modified.

The backend employs the Mistral LLM to generate the recommended positive filters, whereas the filters were previously generated by simple URL parsing and extraction of keywords. Using an LLM allows for a better understanding of web pages based on URLs, and this change is made to generate more accurate and contextually relevant keywords based on the profile entries.

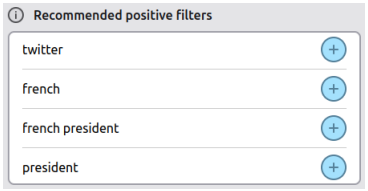

Figure 5.2: Four examples of recommended positive filters generated by the OPP tool.

Upon receiving a request from the frontend to add a profile entry, the backend constructs a prompt that contains the URL of the footprint being added and sends it to Mistral through the Ollama API. The prompt is formatted as follows: "*Given this URL: {url}, what are three keywords that someone can search for to find this web page? Give no explanation. Only give 3 keywords divided by a comma in the format 'x, x, x'.*".

The number of recommended positive filters requested in the prompt is limited to three keywords to ensure the list of generated filters is concise and manageable when displayed to the user. In cases where the LLM responds with more than three keywords, only the first three are kept.

Mistral is given no web page content during the process of generating the recommended positive filters, meaning it only uses the URLs it is given. This choice is made to optimize the filter generation time of the OPP tool.

Despite the prompt's specificity, Mistral's responses may vary in format. To address this, the backend performs a keyword extraction process using a regular expression pattern that identifies potential keywords from the responses. This process also includes cleaning the keywords by removing whitespace and other unwanted characters and making sure that a keyword is valid before adding it to the set of recommended positive filters. A keyword is considered invalid if it is an empty string, exceeds three words, contains underscores, or is outside the length range of 2 to 20 characters. Numeric strings longer than five digits are also considered invalid (i.e. "2024" is valid but "18932714326" is not). These validation steps ensure that the positive filters presented to the user are clear and concise. Sets are used for this part of the implementation to ensure uniqueness for the generated filters.

In cases where Mistral's responses include explanations in addition to a list of keywords, the backend recognizes the explanation as an invalid response because of its length and excludes it. In addition, responses formatted as bullet points or started with for example "Keywords:" are also handled. In the former case, the bullet point character is removed, and in the latter case, the tool will only consider words after the colon.

5.3 Clustering

When a footprint is added to the user profile, the created profile entry is automatically assigned to one of 13 clusters. Inspired by existing categorizations of internet user activities proposed in internet surveys and literature [6, 7, 8, 9], we decided to use the following eight cluster labels: Work, Education, News, Sports, Commercial, Social Media, Entertainment, and Blog. We proposed five additional cluster labels to cover cases not addressed by the literature: Registry, Forum, Encyclopedia, Repository, and Other. These cases were identified by asking the question "During what online activities can other users find information about me?". Such activities could for example be looking someone up through a public registry such as Opplysningen 1881, reading comments on forums such as Stack Overflow, or accessing someone's public GitHub repository.

Table 5.1: The 13 clusters and an example web page for each.

#	URL	Cluster
1	https://en.uit.no/velkommen-som-ansatt	Work
2	https://uit.no/utdanning/program/279506/informatikk_sivilingenior_-_master	Education
3	https://www.theguardian.com/world/emmanuel-macron	News
4	https://olympics.com/ioc/paris-2024	Sports
5	https://www.elkjop.no/	Commercial
6	https://twitter.com/EmmanuelMacron/status/1164617008962527232?lang=en	Social Media
7	https://www.youtube.com/watch?v=9pEqyr_uT-k	Entertainment
8	https://jvns.ca/	Blog
9	https://www.1881.no/	Registry
10	https://stackoverflow.com/questions/13239368/how-to-close-git-commit-editor	Forum
11	https://en.wikipedia.org/wiki/Emmanuel_Macron	Encyclopedia
12	https://github.com/OnlinePrivacyPilot/OnlinePrivacyPilot	Repository
13	https://europa.eu/youreurope/citizens/consumers/internet-telecoms/data-protection-online-privacy/	Other

The clustering is performed with the use of the Mistral LLM. For each user profile entry, a request is sent through the Ollama API to the LLM containing the web page URL and the 13 cluster options. The prompt used for the Ollama request is: "*Based on this web page URL: {url}, what would you cluster this web page as out of the following clusters: {clusters}. Only give the name of the web page's cluster as an answer, give no explanation.*".

The clustering results are presented to the user in the user profile, as shown in Figure 5.3. The figure shows a populated user profile with automatically clustered entries. The profile entries were found by using Emmanuel Macron as the search target, as he is a well-known, public figure.

Users can also choose to manually change the profile entries' clusters to correct the OPP tool's automatic clustering through a drop-down menu containing the 13 clusters, as displayed on the left side of Figure 5.4.

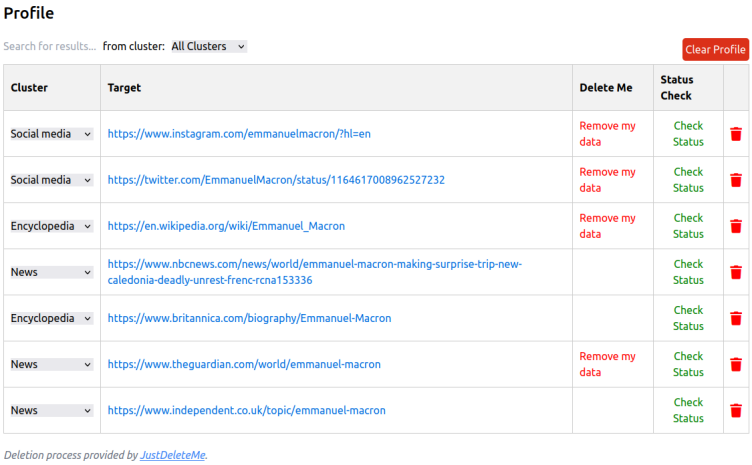

Figure 5.3: Populated user profile with automatically clustered entries.

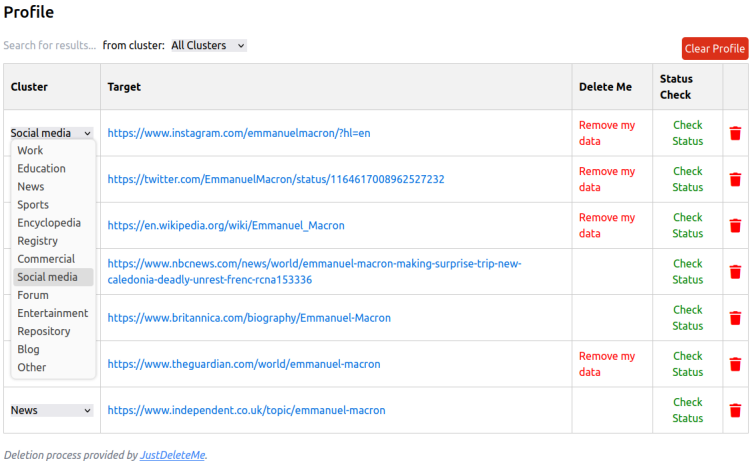

Figure 5.4: Drop-down menu for manually changing a profile entry's cluster.

5.4 Snippets

When a user clicks on an entry in the user profile, a snippet of the web page belonging to that entry is displayed in a modal. Previously, the snippets were generated when the user clicked on an entry in the user profile, which caused a large time overhead when looking through the profile entries. To avoid this, the snippets are now generated by the backend of the OPP tool when the entry is added to the user profile and stored per entry in the database.

The snippet generation process involves sending a request through the Ollama API to Mistral containing the respective web page's URL and the first part of the page's content. The following prompt is sent to Mistral: "*Given this URL:* {*url*},

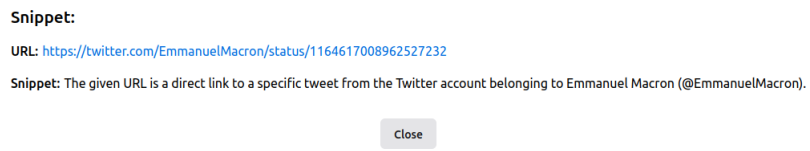

Figure 5.5: Example of a snippet generated for a post by Emmanuel Macron on X.

where the following is the first part of the web page's visible content: '{first_part}'. Generate a summary for this web page. Give no explanation, only respond with the snippet.".

Content extraction for the snippet generation varies based on the web page's type. The PyPDF2 Python library is used to read the contents of PDFs, while the BeautifulSoup Python library parses HTML content. For HTML pages, the visible text is retrieved by filtering out non-content tags and comments. The amount of the web page's content that is provided to Mistral is set to 500 characters to find a balance between efficiency and providing sufficient context for generating helpful snippets.

In cases where the web page content retrieval is unsuccessful, resulting in an empty content string being included in the prompt sent to Mistral, an alternative prompt is used. This prompt is also used when dealing with URLs belonging to X (formerly Twitter), since scraping content from their site is against their Terms of Service, and a paid developer account is needed to read content through their API. The following prompt is used for these cases: "*Given this URL: {url}. Shortly write what this website is based only on the given URL. Do not make up what the content of the web page is. Give no explanation.*". This alternative prompt is needed as Mistral could hallucinate in some cases where the original prompt included an empty content string, and it specifies to Mistral that it should not make up the web page's content and only base the snippet on the given URL.

As Mistral might format its responses differently between runs, each line of the model's response is added to the snippet only if it is determined to be valid. A line is valid if it does not end with a colon character (":"). This is to avoid lines such as "Sure! Here is a snippet for the web page:" or "Summary:" being added to the final snippet.

After being generated by the backend, the snippet is returned to the frontend and displayed to the user in a modal, as shown in Figure 5.5. The figure shows a snippet generated for a post made by Emmanuel Macron's X account.

6 Evaluation

The following experiments have been performed on an HP EliteDesk 800 GB Small Form Factor PC running Ubuntu 22.04.3 LTS (Jammy Jellyfish) with an 11th Gen Intel Core^{TM} i7-11700 2.50 GHz × 16 processor and 16.0 GiB of memory. The LLMs Mistral 7B and Llama 2 7B are used for this evaluation, meaning that both models have seven billion parameters, and are run locally using Ollama.

The evaluation is divided into three sections: recommended positive filter generation, profile entry clustering, and snippet generation.

6.1 Recommended Positive Filters

The evaluation of the recommended positive filter generation involved running the program with a set of predefined URLs to assess the performance of Mistral and Llama 2 in generating relevant keywords. This evaluation is divided into three parts: the LLMs are evaluated in terms of the consistency of generated keywords, and how the usage of three different prompts affects these results. Lastly, Mistral and Llama 2 are compared with each other in terms of how relevant the generated keywords are for a given URL. The 10 URLs listed in Table 6.1 are used for this evaluation. No web page content was given to the LLMs, so all generated keywords are based on the URLs only.

During testing, it was observed that Mistral occasionally returned more than the specified three keywords. For example, Mistral generated eight keywords for one iteration of URL 5 with the third prompt. For these cases, we only included the first three keywords from the response in the results to avoid making the list of recommended positive filters too long when displaying it to the user.

Table 6.1: The 10 URLs used for the recommended positive filter generation evaluation.

#	URL
1	https://en.wikipedia.org/wiki/Emmanuel_Macron
2	https://www.theguardian.com/world/emmanuel-macron
3	https://twitter.com/EmmanuelMacron/status/1164617008962527232?lang=en
4	https://olympics.com/ioc/paris-2024
5	https://en.uit.no/velkommen-som-ansatt
6	https://uit.no/utdanning/program/279506/informatikk_sivilingenior_-_master
7	https://www.youtube.com/watch?v=9pEqyr_uT-k
8	https://stackoverflow.com/questions/13239368/how-to-close-git-commit-editor
9	https://www.1881.no/
10	https://github.com/OnlinePrivacyPilot/OnlinePrivacyPilot

6.1.1 Prompt 1

The first prompt evaluated was "*Given this URL: {url}, what are some keywords that can be extracted? Give no explanation. Only give 3 keywords divided by a comma in the format 'x, x, x'*.". Mistral and Llama 2 were asked to generate keywords for the 10 URLs in Table 6.1, with three iterations each.

Table 6.2 shows that Mistral's keyword generation was mostly consistent, with minor variations, such as in the iterations of URL 5. For half of the URLs, Mistral generated identical keywords across all three iterations.

In contrast, Llama 2 was less consistent between URL iterations compared to Mistral, as indicated by the results in Table 6.3. While Llama 2 consistently generated the same keywords for URLs 4 and 8, its responses for other URLs varied significantly. For example, Llama 2 generated eight distinct keywords for URL 9, with "Norway" being the only repeated keyword between iterations. For URL 5, the first iteration had no valid keywords, resulting in no filters. For iterations two and three, Llama 2 generated six unique keywords, with no overlap between the two sets of keywords. Overall, Llama 2 generated identical keywords for all three iterations for only two of the ten URLs.

Table 6.2: Recommended positive filters generated by Mistral for 10 URLs using the first prompt.

URL	Iteration 1	Iteration 2	Iteration 3
#1	Emmanuel Macron, French politics, President of France	Emmanuel Macron, Politics, President of France	Emmanuel Macron, Politics, President of France
#2	Emmanuel Macron, The Guardian, World	Emmanuel Macron, Theguardian, World	Emmanuel Macron, The Guardian, World
#3	Emmanuelmacron, Status, Twitter	Emmanuelmacron, Status, Twitter	Emmanuelmacron, Status, Twitter
#4	Ioc, Olympics, Paris-2024	Ioc, Olympics, Paris-2024	Ioc, Olympics, Paris-2024
#5	University of Tromsø	Ansatt, UiT NO, Welcome	Ansatt, En, Welcome
#6	Master, Program, Utdanning	Informatikk, Master, Sivilingenior	Informatikk, Master, Sivilingenior
#7	Video, Watch, Youtube	Video, Watch, Youtube	Video, Watch, Youtube
#8	Close, Commit editor, Git	Close, Commit editor, Git	Close, Commit editor, Git
#9	1881, Home, Norway	1881, Furniture, Norway	1881, Homepage, Norway
#10	Github, Onlineprivacypilot, Repository	Github, Onlineprivacypilot, Repository	Github, Onlineprivacypilot, Repository

Table 6.3: Recommended positive filters generated by Llama 2 for 10 URLs using the first prompt. A dashed (-) cell means that no keywords were valid.

URL	Iteration 1	Iteration 2	Iteration 3
#1	France, Macron, Politician	France, Macron, President	Emmanuel Macron, European Union, French politics
#2	Emmanuel Macron, France, World	Emmanuel Macron, France, World	France, Macron, Politics
#3	France, Macron, Twitter	France, Macron, Politics	France, Macron, Tweet
#4	2024, Olympics, Paris	2024, Olympics, Paris	2024, Olympics, Paris
#5	-	Job, Norway, Welcome	Ansatt, Som, Velkommen
#6	Informatikk, Master, Sivilingenior	Informatikk, Master, Sivilingenior	Informatikk, Master, Sivilingeniør
#7	Video, Youtube	Dance, Music	Video, Watch, Youtube
#8	Commit, Editor, Git	Commit, Editor, Git	Commit, Editor, Git
#9	Clothing, Fashion, Shoes	History, Museum, Norway	Bank, Finance, Norway
#10	Online, Pilot, Privacy	Github, Onlineprivacypilot, Privacy	Github, Onlineprivacypilot, Privacy

6.1.2 Prompt 2

The second prompt evaluated was "*Given this URL: {url}, what are three keywords that can be used to describe this web page? Give no explanation. Only give 3 keywords divided by a comma in the format 'x, x, x'.*". Mistral and Llama 2 were again asked to generate keywords for the 10 URLs in Table 6.1, with three iterations each.

With this prompt, Mistral was more consistent when generating keywords compared to the first prompt, as shown in Table 6.4. The most notable inconsistencies were observed for URL 6, with "Master degree" being the only recurring keyword across the first two iterations. For six of the ten URLs, Mistral generated identical keywords across all three iterations. Where differences occurred, they were mostly minor compared to the variations observed with the first prompt.

Llama 2 did not have the same increase in consistency compared to the first prompt as Mistral did. As shown in Table 6.5, the keywords generated by Llama 2 varied significantly between iterations for some URLs. Only URL 4 had consistent keywords across all three iterations, which was a decrease in consistency compared to the first prompt. Some variations were minor, such as switching between writing "Sivilingeniør" with a "ø" and an "o" for URL 6, while others were bigger differences, as seen with URLs 3 and 5.

Table 6.4: Recommended positive filters generated by Mistral for 10 URLs using the second prompt.

URL	Iteration 1	Iteration 2	Iteration 3
#1	Emmanuel Macron, French politician, President	Emmanuel Macron, French politician, President	Emmanuel Macron, French politician, President
#2	Emmanuel Macron, France, President	Emmanuel Macron, France, President	Emmanuel Macron, France, President
#3	Emmanuel Macron, Status, Twitter	Emmanuel Macron, Status, Twitter	Emmanuel Macron, Status, Twitter
#4	Ioc, Olympics, Paris-2024	Ioc, Olympics, Paris-2024	2024, Olympics, Paris
#5	Employees, Welcome	Employees, Welcome	Employees, Welcome
#6	Informatik, Master degree, Sivilingenior	Master degree, Utbildning	Master, Or edukasjon, Utdanning
#7	9peqyr, Video, Youtube	9peqyr, Video, Youtube	9peqyr, Video, Youtube
#8	Commit editor, Git, Stack Overflow	Commit editor, Git, Stackoverflow	Commit editor, Git, Stackoverflow
#9	Antiques, Homepage, Norwegian	Antiques, Homepage, Norwegian	Antiques, Homepage, Norwegian
#10	Github, Online privacy, Pilot	Github, Onlieprivacypilot, Repository	Github, Onlieprivacypilot, Repository

Table 6.5: Recommended positive filters generated by Llama 2 for 10 URLs using the second prompt.

URL	Iteration 1	Iteration 2	Iteration 3
#1	Emmanuel Macron, French politics, President of France	Biography, Emmanuel Macron, French politics	France, Politician, President
#2	France, Macron, World	Emmanuel Macron, France, World	Emmanuel Macron, France, World
#3	France, Politics, President	Emmanuelmacron, France, Politics	France, Macron, Twitter
#4	2024, Olympics, Paris	2024, Olympics, Paris	2024, Olympics, Paris
#5	Job opportunities, Norway, University	Jobs, Norway, UiT	Employment, Job opportunities, Norway
#6	Informatikk, Master, Sivilingenior	Informatikk, Master, Sivilingeniør	Informatikk, Master, Sivilingenior
#7	Dance, Music, Viral	Dance, Music, Performance	Dance, Music
#8	Commit, Editor, Git	Commit, Editor, Git	Commands, Editor, Git
#9	Gambling, Norway, Online	Adventure, Norway, Travel	Gambling, Norway, Sportsbook
#10	Github, Onlineprivacypilot, Privacy	Github, Onlineprivacypilot, Privacy	Online, Privacy, Security

6.1.3 Prompt 3

The third and final prompt evaluated was "*Given this URL: {url}, what are three keywords that someone can search for to find this web page? Give no explanation. Only give 3 keywords divided by a comma in the format 'x, x, x'.*". Mistral and Llama 2 were again asked to generate keywords for the 10 URLs in Table 6.1, with three iterations each.

Using the third prompt, Mistral's consistency between iterations decreased compared to the previous prompts, as shown in Table 6.6. The same keywords were generated for all three iterations for only one of the ten URLs. There were also no valid keywords generated for the second iteration of URL 6. However, some of the variations between iterations were minor, such as the misspelling of "Onlineprivacypilot" as "Onlineprivacypilit" in the first iteration of URL 10, or omitting "news" from "World news" in the third iteration of URL 2.

Llama 2 on the other hand, showed improved consistency when using the third prompt compared to the previous prompts, as shown in Table 6.7. For three of the ten URLs, Llama 2 generated identical keywords for all three iterations, an increased consistency compared to the first two prompts. Differences between iterations of the remaining seven URLs varied. For some, such as URL 2, only one or two words were different between iterations, while others, such as URL 8, had more significant differences between iterations. Despite this, we overall observed a higher consistency from Llama 2 with the third prompt.

Table 6.6: Recommended positive filters generated by Mistral for 10 URLs using the third prompt. A dashed (-) cell means that no keywords were valid.

URL	Iteration 1	Iteration 2	Iteration 3
#1	Emmanuel Macron, French politician, President	Emmanuel Macron, French President, Politician	Emmanuel Macron, French president, Politics
#2	Emmanuel Macron, The Guardian, World news	Emmanuel Macron, The Guardian, World news	Emmanuel Macron, The Guardian, World
#3	Emmanuel Macron, Twitter	Emmanuelmacron, Twitter	Emmanuelmacron, Twitter
#4	Ioc, Olympics, Paris-2024	Ioc website, Paris 2024 Olympics, Paris Olympic games	Ioc, Olympics, Paris 2024
#5	Employees, Norway, UiT	Employment, UiT, UiT website	Employees, UiT website, Welcome page
#6	Informatikk, Master, Sivilingenior	-	Informatikk master, Sivilingenior master
#7	9peqyr, Watchvideo, Youtube	9peqyr, Watch, Youtube	9peqyr, Watch video, Youtube
#8	Close, Commit editor, Git	Close, Commit editor, Git	Close, Commit editor, Git
#9	1881 website, Norwegian company, Tech solutions	1881, Norwegian fashion, Online store	1881 website, Home decoration, Norwegian retail
#10	Github, Onlineprivacypilit, Repository	Github, Onlineprivacypilot, Repository	Github, Onlineprivacypilot, Repository

Table 6.7: Recommended positive filters generated by Llama 2 for 10 URLs using the third prompt.

URL	Iteration 1	Iteration 2	Iteration 3
#1	Emmanuel Macron, French politics, President of France	Emmanuel Macron, French president, Political leader	Emmanuel Macron, French politics, President of France
#2	Emmanuel Macron, France, President	Emmanuel Macron, France, Politics	Emmanuel Macron, France, Politics
#3	Emmanuel Macron, France, President	Emmanuel Macron, France, Politics	France, Macron, President
#4	2024, Olympics, Paris	2024, Olympics, Paris	2024, Olympics, Paris
#5	Ansatt, Norway, Velkommen	Job opportunities, Norway, Welcome	Ansatt, UiT Norway, Velkommen
#6	Informatikk, Master, Sivilingeniør	Informatikk, Master, Sivilingeniør	Informatikk, Master, Sivilingeniør
#7	Video, Watch, Youtube	Video, Watch, Youtube	Video, Watch, Youtube
#8	Close git commit, Commit editor close, Git commit editor	Commit, Editor, Git	Close git commit, Git commit editor, Git editor close
#9	Genealogy, History, Norway	1881, History, Norway	Historical, Maps, Norway
#10	Github, Onlineprivacypilot, Privacy	Github, Online privacy pilot, Online security	Github, Onlineprivacy, Pilot

6.1.4 Relevance of keywords

This section evaluates the relevance of the keywords generated by Mistral and Llama 2 for the 10 URLs in Table 6.1.

Looking at the keywords generated by Mistral listed in tables 6.2, 6.4, and 6.6, we can see that Mistral was able to generate relevant keywords for most of the URLs across the three prompts. However, URL 9 was more challenging, with the relevant keyword "1881" only being included when using the first and third prompts.

For URL 7, a URL for a YouTube video, Mistral frequently included "9peqyr" in its keywords, which is part of the video's identifier. Only with prompt 1 did Mistral not include this as one of the keywords. Aside from "9peqyr", Mistral generated relevant keywords such as "Video", "Watch", and "YouTube" for this URL.

Llama 2 was also able to generate relevant keywords for most of the URLs across the three prompts, as shown in tables 6.3, 6.5, and 6.7. However, for URL 9, Llama 2 only generated one relevant keyword across all three prompts, including the keyword "1881" only for the second iteration with the third prompt.

For URL 7, Llama 2 sometimes generated general YouTube-related keywords like "Dance" and "Music". While these words may be associated with YouTube content in general, they are not specific to the video in question. Given that no

web page content was provided for context, it would have been preferable if Llama 2 had focused on more relevant keywords such as "Video" and "YouTube". Like Mistral, Llama 2 included the video identifier "9pEqyr_uT-k" in some iterations, but this was correctly filtered out during the processing of the model's response.

Overall, both models generated keywords with similar levels of relevance for eight of the ten URLs. However, Mistral more frequently included keywords such as "Twitter", "Stack Overflow", and "YouTube" for the respective URLs, which indicates a slight increase in relevance over Llama 2.

6.1.5 Processing the LLMs' responses

The method of extracting the keywords from the LLMs' responses also affects the results discussed above. A regular expression pattern was used to identify keywords in the responses, which occasionally led to non-keyword matches when the LLMs included additional text along with the keywords.

It was observed that both LLMs often included additional text in their responses. Mistral sometimes included multiple sets of keywords divided by "Or:" or added "Note:" followed by notes in its responses, while Llama 2 typically formatted its responses as a phrase such as "*Sure! Here are three keywords that someone can use to find the web page you provided: "YouTube, video, watch*"". Although these consistent formats were handled during the keyword extraction process, we still observed some unexpected formats in the LLMs' responses. One such example is the previously mentioned case where both LLMs for some iterations of URL 7 would include the video identifier of the YouTube URL as a keyword. In Llama 2's case, the keyword "9pEqyr_uT-k" was correctly excluded during the extraction process, but not for Mistral, as seen in tables 6.4 and 6.6 where the keyword "9peqyr" is included. This happened because Mistral added a backslash to the original part of the video identifier when including it in its response ("9pEqyr_uT-k" instead of "9pEqyr_uT-k"), causing the word to be split into a part that did not get excluded during extraction.

On two occasions, no keywords were extracted from the LLMs' responses. This was either due to invalid keywords or the response's format not being recognized. The first case happened during Llama 2's first iteration on URL 5 using the first prompt. Llama 2's response was simply "x, x, x", resulting in no keywords, as shown in Table 6.3.

The second case where no keywords were extracted was for Mistral's second iteration on URL 6 using the third prompt. In this case, none of the keywords were valid as they all either exceeded the character or word count, resulting

in no returned keywords, as shown in Table 6.6.

6.1.6 Conclusion

In summary, the evaluation of the recommended positive filter generation indicates that Mistral was generally more consistent than Llama 2, particularly with the second prompt. While both models were similar in the relevance of the generated keywords, Mistral's keywords were slightly more relevant, especially for URLs 7 and 9. Considering both the overall relevance and consistency of the generated positive filters, Mistral was selected as the preferred LLM over Llama 2. Despite the third prompt being the least consistent for Mistral, these inconsistencies were relatively minor. Additionally, the third prompt gave the most relevant keywords overall. It was therefore decided to use Mistral with the third prompt for the implementation of the recommended positive filters feature.

6.2 Clustering

The evaluation of the user profile entry clustering focuses on two main areas. The LLMs Mistral and Llama 2 are compared in terms of speed and number of correct clusters. Then we look at the effects that varying the amount of content provided to the LLMs for context has on the clustering correctness and speed for the two models.

The clustering evaluation was performed using a set of 13 URLs, which are presented in Table 6.8. These URLs were selected to represent a diverse range of possible digital footprints and to test the clustering capabilities of the LLMs for different clusters. The first four URLs were found by using Emmanuel Macron as the search target in the OPP tool to see how the LLMs handle potential real use cases. The remaining nine URLs were chosen to cover a variety of clusters, to evaluate how the LLMs perform on a broader set of web page types.

Table 6.9 shows the results when asking Mistral and Llama 2 to cluster the 13 URLs five times each without providing them with any of the web pages' contents. As seen in the table, Mistral was able to cluster most of the URLs correctly, correctly clustering them 54 out of 65 times, while Llama 2 only clustered the URLs correctly 39 out of 65 times. On the other hand, while Mistral had the highest clustering correctness, Llama 2 clustered the URLs faster on average.

Table 6.10 shows the results of asking Mistral and Llama 2 to cluster the same

Table 6.8: The 13 URLs used for the clustering evaluation and their predefined clusters.

#	URL	Cluster
1	https://en.wikipedia.org/wiki/Emmanuel_Macron	Encyclopedia
2	https://www.theguardian.com/world/emmanuel-macron	News
3	https://twitter.com/EmmanuelMacron/status/1164617008962527232?lang=en	Social Media
4	https://olympics.com/ioc/paris-2024	Sports
5	https://en.uit.no/velkommen-som-ansatt	Work
6	https://uit.no/utdanning/program/279506/informatikk_sivilingenior_-_master	Education
7	https://www.youtube.com/watch?v=9pEqyr_uT-k	Entertainment
8	https://stackoverflow.com/questions/13239368/how-to-close-git-commit-editor	Forum
9	https://www.1881.no/	Registry
10	https://github.com/OnlinePrivacyPilot/OnlinePrivacyPilot	Repository
11	https://www.elkjop.no/	Commercial
12	https://jvns.ca/	Blog
13	https://europa.eu/youreurope/citizens/consumers/internet-telecoms/data-protection-online-privacy/	Other

Table 6.9: Clustering with no web page content provided. Columns show the number of correct clusters and average time per clustering out of 65 runs.

Model	Correctly Clustered	Avg. Clustering Time (s)
Mistral	54/65	5.23s
Llama 2	39/65	2.90s

13 URLs another five times each, but this time providing them with the first 250 characters of the web pages' visible content in addition to the URL. The choice of including the first 250 characters is made on the assumption that the first content of a web page often includes relevant keywords or information that could help in the clustering process. However, the results indicate that both models determined the correct cluster fewer times when given this small amount of context than they did with no web page content at all. This outcome suggests that the provided content may not have contained enough relevant information or could have included content that gave the LLMs the wrong context for the URLs, leading to confusion for the LLMs.

As an example, if the first 250 characters of an employee page for a university contains information about the university as a whole and not specifically about being employed there, it may not provide sufficient context for the LLMs to identify that employees use the web page and therefore belongs to the work cluster. This can lead to the LLMs clustering the web page as education instead of work, which we saw in the case of URL 5.

In addition, as the web page content had to be retrieved and considered for the clustering, the process took longer on average than with no content.

After seeing the negative effects of giving only a small amount of web page content in addition to the URLs had on the clustering performance, the amount of content was increased to 500 characters. The results of asking the models

Table 6.10: Clustering with 250 characters of web page content provided. Columns show the number of correct clusters and average time per clustering out of 65 runs.

Model	Correctly Clustered	Avg. Clustering Time (s)
Mistral	43/65	6.90s
Llama 2	30/65	6.24s

to cluster the same 13 URLs, five times each, are displayed in Table 6.11. These results show an improvement in the number of correctly clustered URLs compared to when providing only 250 characters of content, but it was still lower than when no content was provided. In addition, the average time spent on each clustering continued to increase and was now more than three and almost 4 seconds longer than when no content was given for Mistral and Llama 2 respectively.

Table 6.11: Clustering with 500 characters of web page content provided. Columns show the number of correct clusters and average time per clustering out of 65 runs.

Model	Correctly Clustered	Avg. Clustering Time (s)
Mistral	48/65	8.40s
Llama 2	31/65	6.53s

Seeing as the number of correctly clustered URLs increased between providing 250 characters and 500 characters of web page content to the LLMs, we tried increasing this amount further to 1000 characters of web page content to see if the results would continue to climb. As seen in Table 6.12 however, Mistral determined fewer clusters correctly than it did with 500 characters of content and Llama 2 determined the same number of correct clusters. The average time taken for each clustering also continued to increase for both models.

Table 6.12: Clustering with 1000 characters of web page content provided. Columns show the number of correct clusters and average time per clustering out of 65 runs.

Model	Correctly Clustered	Avg. Clustering Time (s)
Mistral	44/65	12.96s
Llama 2	31/65	11.02s

As we saw that both LLMs performed better in the case where none of the web pages' contents were provided, we took a closer look at how the models did for each of the 13 URLs listed in Table 6.8 when given no content (i.e. only the URLs of the web pages were given as context).

Table 6.13 shows how many times out of five runs Mistral and Llama 2 correctly clustered the 13 URLs. Both models were able to correctly cluster URLs 1-4, 6, and 11 for all five runs, but got varying results for the remaining seven URLs. Overall, Mistral clustered the most URLs correctly, except for URL 5 which belongs to the "Welcome as an employee" page on UiT's website. This is however a tricky case because the URL belongs to a university website, so Mistral clustered the URL as education on all five runs. Llama 2, on the other hand, correctly clustered this URL as work all five times. The two remaining URLs that Mistral assigned to the wrong clusters were URL 9 which belongs to the Norwegian public registry website Opplysningen 1881, and URL 13 which is the European Union's web page for "Data protection and online privacy". Four of five times Mistral clustered URL 9 as registry, which is the correct cluster. The fifth time it clustered it as commercial. For URL 13, Mistral clustered the URL as registry all five times instead of other.

For URL 7, Llama 2 determined the correct cluster of entertainment four of five times but assigned it to the education cluster the final time. Llama 2 did not cluster URLs 8-10, 12, and 13 correctly in any of the five runs.

Table 6.13: Mistral and Llama 2 clustering correctness when only providing the URL and no web page content. Results are given as how many of five runs per URL the models clustered correctly.

Model	URL 1	URL 2	URL 3	URL 4	URL 5	URL 6	URL 7	URL 8	URL 9	URL 10	URL 11	URL 12	URL 13
Mistral	5/5	5/5	5/5	5/5	0/5	5/5	5/5	5/5	4/5	5/5	5/5	5/5	0/5
Llama 2	5/5	5/5	5/5	5/5	5/5	5/5	4/5	0/5	0/5	0/5	5/5	0/5	0/5

Table 6.14 shows the average time (in seconds) of five runs that Mistral and Llama 2 used when clustering the same 13 URLs. As seen from the results, both models take a few seconds on average to cluster the URLs. While Llama 2 is more consistently around 2-4 seconds, Mistral has average times spanning from as little as 2.35s for URL 7 to 14.98s for URL 9. Only for four of the URLs did Mistral get an average time that is lower than Llama 2's results.

Table 6.14: Clustering speed in seconds for Mistral and Llama 2 when only providing the URL and no web page content. Times are given as the average of five runs per URL.

Model	URL 1	URL 2	URL 3	URL 4	URL 5	URL 6	URL 7	URL 8	URL 9	URL 10	URL 11	URL 12	URL 13
Mistral	5.62s	5.10s	3.06s	3.84s	6.47s	2.54s	2.35s	2.50s	14.98s	2.39s	6.00s	4.63s	8.50s
Llama 2	2.96s	3.37s	3.11s	2.22s	2.29s	2.58s	2.55s	2.50s	3.29s	4.08s	3.29s	2.96s	2.50s

During the testing of Llama 2's clustering, we saw that the model would overguess the work and education categories, no matter the order of the URLs. As an example, for URLs 12 and 13 (see Table 6.8) when not using any web page content, Llama 2 clustered the URLs as education for all five iterations on each, instead of blog and other, respectively. This happened even if the URL

belonging to education was clustered after these two URLs. For URL 8, Llama 2 clustered it as work the first time, and education the remaining four times while its predefined cluster is forum. As the content length provided to the LLMs increased, we also observed that the number of times Llama 2 assigned URLs to the work cluster increased.

While this means that Llama 2 will guess URLs belonging to the education and work clusters correctly more often than Mistral will through over-guessing these clusters, Mistral still clusters the most URLs correctly.

In conclusion, the clustering evaluation shows a trade-off between clustering correctness and time efficiency. Mistral achieved higher clustering correctness, while Llama 2 was faster on average. Given the importance of clustering correctness for our use case, Mistral was selected as the preferred LLM for the clustering feature with no provided web page content.

6.3 Snippets

The evaluation of the snippet generation involves seeing how providing Mistral and Llama 2 with different amounts of web page content affects the content of the snippets they generate. The eight URLs in Table 6.15 are used for this evaluation, and the models were asked to generate snippets of each URL three times each. The models were first given no web page content, followed by three different amounts of web page content: 500, 1000, and 2000 characters.

Table 6.15: The eight URLs used for the snippet evaluation.

#	URL
1	https://twitter.com/EmmanuelMacron/status/1164617008962527232?lang=en
2	https://olympics.com/ioc/paris-2024
3	https://en.uit.no/velkommen-som-ansatt
4	https://uit.no/utdanning/program/279506/informatikk_sivilingenior_-_master
5	https://www.youtube.com/watch?v=9pEqyr_uT-k
6	https://stackoverflow.com/questions/13239368/how-to-close-git-commit-editor
7	https://www.1881.no/
8	https://github.com/OnlinePrivacyPilot/OnlinePrivacyPilot

As an example, these are the snippets that Mistral generated during its first iteration of URL 2 with varying content lengths:

- **No web page content:** "*The URL <https://olympics.com/ioc/paris-2024> leads to the official website for the Paris 2024 Olympic Games, organized by the International Olympic Committee (IOC), as indicated by the "ioc" part*

of the URL.".

- **500 characters of web page content:** "*The Olympics Museum, Athlete365 Olympic Studies Centre, and Olympic Refuge Foundation are introductions to the Paris 2024 Olympic Games. Latest news includes stories about the Refugee Olympic Team receiving Bulgarian citizenship for member Kimia Alizadeh, and the Paris 2024 IOC launching an innovative mobile game. Atos expresses excitement for their involvement in the Paris 2024 Olympics.*".

- **1000 characters of web page content:** "*The Olympics.com webpage for Paris 2024 covers news and updates on the upcoming Olympic Games, including articles on athlete stories, the refugee team, innovations such as the new mobile game, and ceremonies like the lighting of the Olympic flame. Additionally, there are sections for museum visits, athlete resources, and shopping.*".

- **2000 characters of web page content:** "*The webpage is about the Paris 2024 Olympics with latest news, athlete stories, and information on various initiatives such as the Refugee Olympic Team, Paris 2024 mobile game, Olympic flame, Deloitte campaign, starting line for athletes, World Health Day, Olympic Aquatics Centre, gender equality WISH programme, and sustainability efforts. The IOC provides access to key documents, media guidelines, and news access rules. The page also includes sections for fans to learn about the athletes and sports.*".

Llama 2 generated the following snippets for its first iteration of the same URL with varying content lengths:

- **No web page content:** "*The website at the provided URL is the official website of the International Olympic Committee (IOC) for the 2024 Summer Olympics, which are scheduled to take place in Paris, France.*".

- **500 characters of web page content:** "*Olympics Museum Athlete365 Olympic Studies Centre Olympic Refuge Foundation Paris 2024 Shop English IOC Olympic Games Athletes Olympic Movement Beyond the Games Media News Documents FAQ Paris 2024 Latest news Latest news Paris 2024 Atos on the Road to Paris: “These Games are very close to our heart*"".

- **1000 characters of web page content:** "** IOC launches innovative mobile game ahead of Olympic Games* Refugee Olympic Team member receives Bulgarian citizenship* Paris 2024 Olympic flame lit in symbolic ceremony in Ancient Olympia* Deloitte celebrates pioneering "firsts" in new Olympic campaign* Athletes embark on a journey to rediscover the people and places that fueled their passion* World Health Day celebrated, highlighting the*

positive impact of sport and physical activity in Paris and beyond.".

- **2000 characters of web page content:** "** Preparations for the Olympic Games and the IOC's support for the Organising Committee* Innovative initiatives and campaigns, such as the Paris 2024 mobile game and the "Paris je t'aime" exhibition on sport and urban cultures* Sustainability efforts, including partnerships with social and solidarity companies* Gender equality programs, such as the WISH program for female coaches* Updates on qualification information, anti-doping rules, and medical procedures* Access to key documents, such as the Host City Contract and media guidelines.*".

During the evaluation, we observed that Mistral and Llama 2's responses were consistently formatted across most iterations. For the tests with no web page content, the differences between the two models' responses were small. However, without any web page content, the models sometimes lacked the necessary context to provide accurate information in their snippets. For example, when given no content for URL 4, neither Mistral nor Llama 2 correctly identified that the URL belonged to the University of Tromsø, instead associating it with other universities such as the University of Stavanger, the University in Trondheim, or the University of Agder. This did not happen for URL 2 in the examples above, as the models were able to get sufficient context from the URL alone, but it shows the models' limitations in recognizing the correct context for URLs that include unfamiliar words or phrases without additional content.

As the amount of content provided increased, Mistral consistently generated natural snippets, while also summarizing the given content. Llama 2 on the other hand, tended to list the actual contents of the web pages by extracting parts of the given content. This is particularly apparent in the examples above for URL 2. This simple listing of content did not happen for all iterations, but overall, we saw that Mistral generated more natural snippets than Llama 2. As a result, we selected Mistral for the snippet generation feature of the OPP tool. To find a balance between efficiency and providing sufficient context when generating the snippets, we decided to give Mistral 500 characters of web page content.

After incorporating Mistral into the snippet generation feature of the OPP tool, we tested it by adding footprints related to Emmanuel Macron to the user profile. When looking at the snippets generated by Mistral for these entries, we observed that the model could sometimes hallucinate, generating incorrect or fabricated snippets for some web pages. For example, it would occasionally make up a random number of followers for Emmanuel Macron when generating a snippet for his Instagram page, or create fake news articles when generating snippets for BBC. This issue occurred in cases where the tool failed to retrieve the content of the web page in question, resulting in an empty string in the

prompt sent to Mistral. Since the prompt requested a snippet based on both the URL and the web page content, Mistral made up the content of the web pages itself in the absence of their actual content. We then tested Llama 2 under the same conditions and saw that it also hallucinated for the same cases as Mistral.

To address this issue, we added the following alternative prompt for cases where the web page content string is empty: "*Given this URL:* {*url*}. *Shortly write what this website is based only on the given URL. Do not make up what the content of the web page is. Give no explanation.*". With this addition, both Mistral and Llama 2 stopped generating the fabricated content and instead based their snippets only on the URLs in cases where the tool could not retrieve the web page content, as specified by the prompt.

7 Discussion

7.1 Using Other LLMs

7.1.1 ChatGPT

In addition to Llama 2 and Mistral, other models were also considered when incorporating LLMs into the OPP tool, such as ChatGPT. The recommended positive filter generation and clustering features were tested through OpenAI's web version of ChatGPT, and it showed promising results when asked to perform these tasks.

We did however not end up using ChatGPT as it is not open-source and does not offer free access to its API. This means that the user of the OPP tool would have to create an OpenAI account and pay for access to the API [50] on the OpenAI Developer Platform to gain access to models such as their GPT-3.5 Turbo model.

7.1.2 Llama 3

Meta's newest model and Llama 2's successor, Llama 3, was released in April 2024 and was therefore not considered as an option for the OPP tool. It would however be interesting to see how it compares to the other models in the future, especially as it has displayed an increase in performance from its predecessor, Llama 2 [42].

7.2 X

Previously, the OPP tool was able to retrieve data from X (formerly Twitter), but with recent changes to their Terms of Service and account creation, one would now have to create and pay for a developer account to read content through their API. Bypassing their API and scraping the site directly is also not an option as it is against their Terms of Service. While they do offer a free plan for access to the API, this version only provides write access to the API [51]. Therefore, to handle cases where X is the source of a user's footprint, the OPP tool no longer retrieves content from their site, which means that Mistral will only consider the URLs when generating the respective snippets.

In addition to X, a problem with retrieving information from Instagram was encountered. While the OPP tool previously retrieved a user's name, username, and description from their Instagram profile to display in the result graph, it is no longer able to do so. These two cases display the changing nature of websites, and especially social media platforms, in terms of accessing information and emphasize the need to properly read through the terms and conditions before performing web scraping or using other methods for retrieving information.

7.3 Ethical and Legal Considerations

The OPP tool aims to give users a better overview of their online fingerprint and guide them through removing their personal information where possible. While the tool uses Google API keys to link users' requests to their Google accounts and aims to help individuals manage their online fingerprint, it can still be used for malicious purposes such as collecting information about another individual. It is therefore important to consider the ethical and legal aspects of both the usage and development of tools such as the OPP tool.

An important consideration regarding the development of the tool is the use of techniques such as web scraping, which presents several legal and ethical concerns. The legal landscape surrounding web scraping is complex and is still evolving, as demonstrated by various court cases. For instance, hiQ Labs legally challenged LinkedIn after receiving a cease and desist order to stop scraping public profile data on their site [29, p. 181-182]. This case highlights the question surrounding what should be considered publicly accessible data and the legal aspects of scraping such data. Similarly, in the case of Ticketmaster vs. Riedel Marketing Group, the latter was found to have infringed on Ticketmaster's copyrighted material by scraping the site for tickets to resell, despite agreeing to the site's terms and conditions [29, p. 182]. These examples highlight how

the use of web scraping must follow various legal frameworks, some of which may be region-dependent, such as the Computer Fraud and Abuse Act (CFAA) in the United States [29, p. 184-185], and to respect the terms and conditions set by websites.

In addition to legal compliance, it is important to consider the ethical implications of using the OPP tool, especially in terms of its potential impact on an individual's privacy. Tools such as the OPP tool must guide users to use them responsibly and for their intended purpose.

7.4 Future Work

7.4.1 Using a combination of the models' results

In the future, it could be helpful to explore ways of combining multiple LLMs' results for the recommended positive filters, clustering, and snippet features since models may perform better for different tasks.

For instance, in the case of clustering, the OPP tool could base the assigned cluster on which model is best at determining certain clusters, for example, by using confidence scores for each cluster. The confidence scores could be determined by testing the models on the different clusters and seeing how often they assign the correct cluster. With this method, however, one would have to handle cases such as the models disagreeing on the assigned cluster while having the same confidence score for their respective cluster.

7.4.2 Status check functionality

The status check functionality of the OPP tool allows the user to check if entries in the user profile are still "alive", meaning that the corresponding information about the user is still publicly available on the web. This status checking is performed by checking the status codes returned by the respective web pages, which leaves many edge cases to cover.

There is therefore a need to improve the process of checking the status of profile entries by exploring alternative ways of performing the status check. This change is needed to better maintain the consistency between the user profile and the current web, and to preserve the soundness of the user profile.

7.4.3 Ethical and legal considerations for user profile storage

There is a need to further explore the ethical and legal aspects of storing the user profile and to incorporate additional steps to guide the user to only store information about themselves, for instance through a Terms of Use agreement or Privacy Policy as is used by the Internet Project [58], or by presenting users with ethical and legal considerations regarding online privacy. Encouraging users of the OPP tool to only use it for its intended purpose is important to keep in mind for further development of the tool.

8 Conclusion

This thesis presents the incorporation of large language models (LLMs) into the Online Privacy Pilot (OPP) tool, an open-source tool for retrieving and displaying a user's digital footprints to help them manage their online fingerprint.

The OPP tool was developed in 2023 by Benoît Leconte and Daniel Nicolas Pressensé during their internship at the Department of Computer Science at UiT. The tool has since been used as a case study for the Capstone project preceding this Master's thesis, where features based on ideas from information retrieval literature were implemented. These features include a user profile, recommended positive filter generation, clustering, snippets, and status check functionalities.

The goal of this thesis was to explore how LLMs can be incorporated for semantic search techniques in information retrieval systems such as the OPP tool and further help users manage their online fingerprint. The main focus was therefore on the incorporation of LLMs into the recommended positive filter, clustering, and snippet features. An additional step was also implemented when adding a footprint to the user profile to guide users to only store footprints related to themselves. A total of 13 cluster labels based on research on online activities were also selected and proposed for the clustering feature of the OPP tool.

From the evaluation of Mistral and Llama 2, it was seen that Mistral displays

better performance for the recommended positive filter, clustering, and snippet features, and it was therefore selected as the preferred model to use for these features in the OPP tool.

Future work includes exploring the possibility of combining the results of multiple LLMs for the implemented features of the OPP tool, and the improvement of the status check functionality of the tool. There is also a need to further explore the legal and ethical aspects of storing the user profile and how it affects user privacy.

The incorporation of LLMs into the OPP tool has displayed improvements for the recommended positive filter, clustering, and snippet features of the tool. By employing Mistral, specifically, the OPP tool can generate semantically similar terms for the list of recommended positive filters, generate helpful snippets for the profile entries, and automatically cluster the footprints as they are added to the user profile. The work done for this thesis allows for further exploration of the incorporation of LLMs into the OPP tool and how LLMs can be employed to help users more effectively manage their online fingerprint.

Bibliography

[1] Danielle F. O. Vik, (2023). *Information Retrieval Techniques for Managing Online Fingerprint*. Unpublished, Dec. 2023.

[2] B. Leconte and D. Pressensé. *OnlinePrivacyPilot*. GitHub. URL: https://github.com/orgs/OnlinePrivacyPilot/repositories. Accessed 10.04.2024.

[3] B. Leconte and D. Pressensé. *OnlinePrivacyPilot, User Documentation*. GitHub. URL: https://github.com/OnlinePrivacyPilot/OnlinePrivacyPilot_UserDoc. Accessed 10.04.2024.

[4] UiT - The Arctic University. *Cyber Security Group (CSG)*. URL: https://uit.no/research/csg. Accessed 12.04.2024.

[5] C. Manning, P. Raghavan, and H. Schütze, (2008). *Introduction to Information Retrieval*. Cambridge University Press, Cambridge. ISBN: 0521865719. URL: https://nlp.stanford.edu/IR-book/.

[6] G. Blank and D. Groselj, (2014). *Dimensions of Internet use: amount, variety, and types*. Information, Communication, and Society, Vol. 17, Issue 4, pp. 417-435. URL: http://dx.doi.org/10.1080/1369118X.2014.889189.

[7] G. Blank, W.H. Dutton, and J. Lefkowitz, (2019). *Perceived Threats to Privacy Online: The Internet in Britain, the Oxford Internet Survey, 2019*. URL: http://dx.doi.org/10.2139/ssrn.3522106.

[8] H. Lebo, (2018). *The World Internet Project International Report (Ninth Edition)*. Center for the Digital Future, USC Annenberg School for Communication and Journalism, University of Southern California. URL: https://www.digitalcenter.org/wp-content/uploads/2019/01/World-Internet-Project-report-2018.pdf.

[9] M. Dunahee and H. Lebo, (2018). *The 2018 Digital Future Report, Surveying the Digital Future, Year Sixteen*. Center for the Digital Future,

USC Annenberg School for Communication and Journalism, University of Southern California. URL: https://www.digitalcenter.org/wp-content/uploads/2018/12/2018-Digital-Future-Report.pdf.

[10] World Internet Project. *World Internet Project (WIP)*. URL: https://www.worldinternetproject.com/about.html. Accessed 14.05.2024.

[11] JustDeleteMe Contrib Group. *JustDeleteMe database*. GitHub. URL: https://github.com/jdm-contrib/jdm/blob/master/_data/sites.json. Accessed 10.05.2024.

[12] R. Knight, E. Poole, and JDM Contrib Team. *JustDeleteMe Website*. URL: https://justdeleteme.xyz/. Accessed 10.05.2024.

[13] R. Lewis, E. Poole, JDM Contrib Team, and JGMD Contrib Team. *JustGetMyData Website*. URL: https://justgetmydata.com/. Accessed 10.05.2024.

[14] R. Lewis, and E. Poole, JDM Contrib Team, JGMD Contrib Team, and JWTD Contrib Team. *JustWhatsTheData Website*. URL: https://justwhatsthedata.github.io/. Accessed 10.05.2024.

[15] Incogni Inc. *Incogni*. URL: https://incogni.com/. Accessed 10.05.2024.

[16] Abine Inc. *DeleteMe*. URL: https://joindeleteme.com/. Accessed 10.05.2024.

[17] Abine Inc., DeleteMe. *Search Yourself*. URL: https://help.joindeleteme.com/hc/en-us/articles/13593247145747-Search-Yourself. Accessed 10.05.2024.

[18] F. Erlandsson, R. Nia, M. Boldt, H. Johnson, and S.F. Wu, (2015). *Crawling Online Social Networks*. Second European Network Intelligence Conference, Karlskrona, Sweden, pp. 9-16. DOI: 10.1109/ENIC.2015.10. URL: https://doi.org/10.1109/ENIC.2015.10.

[19] J. Wieringa, P.K. Kannan, X. Ma, T. Reutterer, H. Risselada, and B. Skiera, (2021). *Data analytics in a privacy-concerned world*. Journal of Business Research, Vol. 122, pp. 915-925. URL: https://doi.org/10.1016/j.jbusres.2019.05.005.

[20] M.M. Nair, and A.K. Tyagi, (2021). *Privacy: History, Statistics, Policy, Laws, Preservation and Threat Analysis*. Journal of Information Assurance and Security. Vol. 16, Issue 1, pp. 24-34.

[21] Y. Hwang, I. Lee, H. Kim, H. Lee, and D. Kim, (2022). *Current Status and Security Trend of OSINT*. Wireless Communications and Mobile Computing, Vol. 2022. Article ID 1290129. DOI: 10.1155/2022/1290129. URL: https://doi.org/10.1155/2022/1290129.

[22] J. Pastor-Galindo, P. Nespoli, F.G. Mármol, and G.M. Pérez, (2020). *The Not Yet Exploited Goldmine of OSINT: Opportunities, Open Challenges and Future Trends*. In IEEE Access, Vol. 8, pp. 10282-10304. DOI: 10.1109/ACCESS.2020.2965257. URL: https://doi.org/10.1109/ACCESS.2020.2965257.

[23] OxIS. *Oxford Internet Surveys*. URL: https://oxis.oii.ox.ac.uk. Accessed 14.05.2024.

[24] W.H. Dutton and G. Blank, (2011). *Next Generation Users: The Internet in Britain*. Oxford Internet Survey 2011. Oxford Internet Institute, University of Oxford. URL: https://oxis.oii.ox.ac.uk/wp-content/uploads/sites/16/2014/11/oxis2011-report.pdf.

[25] Center for the Digital Future. *Reports*. URL: https://www.digitalcenter.org/reports/. Accessed 14.05.2024.

[26] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, W. Ye, Y. Zhang, Y. Chang, P.S. Yu, Q. Yang, and X. Xie, (2024). *A Survey on Evaluation of Large Language Models*. ACM Transactions on Intelligent Systems and Technology, Vol. 15, Issue 3, Article 39, pp. 1-45. URL: https://doi.org/10.1145/3641289.

[27] R. Guha, R. McCool, and E. Miller, (2003). *Semantic search*. In Proceedings of the 12th international conference on World Wide Web (WWW '03). Association for Computing Machinery, New York, NY, USA, pp. 700–709. DOI: 10.1145/775152.775250. URL: https://doi.org/10.1145/775152.775250.

[28] W. Wei, P.M. Barnaghi, and A. Bargiela, (2008). *Search with meanings: an overview of semantic search systems*. International Journal of Communications of SIWN, Vol. 3, pp. 76–82.

[29] S. vanden Broucke, and B. Baesens, (2018). *Practical Web Scraping for Data Science*. Apress, Berkeley, CA. ISBN: 978-1-4842-3581-2. URL: https://doi.org/10.1007/978-1-4842-3582-9.

[30] M.A. Khder, (2021). *Web Scraping or Web Crawling: State of Art, Techniques, Approaches and Application*. International Journal of Advances in Soft Computing and its Applications. Vol. 13. pp. 145-168. DOI: 10.15849/I-

JASCA.211128.11. URL: https://doi.org/10.15849/IJASCA.211128.11.

[31] M. Kobayashi, and K. Takeda, (2000). *Information retrieval on the web*. ACM Computing Surveys, Vol. 32, Issue 2, pp. 144–173. DOI: 10.1145/358923.358934. URL: https://doi.org/10.1145/358923.358934.

[32] M. Hildebrand, J.R. Ossenbruggen, van, and L. Hardman, (2007). *An analysis of search-based user interaction on the semantic web*. CWI report. INS-E, Vol. 0706. Centrum voor Wiskunde en Informatica, Amsterdam, Holland. URL: https://research.tue.nl/en/publications/an-analysis-of-search-based-user-interaction-on-the-semantic-web.

[33] Y. Zhu, P. Zhang, C. Zhang, Y. Chen, B. Xie, Z. Dou, Z. Liu, and J.R. Wen, (2024). *INTERS: Unlocking the Power of Large Language Models in Search with Instruction Tuning*. arXiv preprint arXiv:2401.06532. URL: https://doi.org/10.48550/arXiv.2401.06532.

[34] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, X. Jiang, K. Cobbe, T. Eloundou, G. Krueger, K. Button, M. Knight, B. Chess, and J. Schulman, (2021). OpenAI. *WebGPT: Browser-assisted question-answering with human feedback*. arXiv preprint arXiv:2112.09332. URL: https://arxiv.org/abs/2112.09332.

[35] L. Wang, N. Yang, and F. Wei, (2023). *Query2doc: Query Expansion with Large Language Models*. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 9414–9423, Singapore. Association for Computational Linguistics. URL: https://doi.org/10.18653/v1/2023.emnlp-main.585.

[36] V. Rawte, A. Sheth, and A. Das, (2023). *A survey of hallucination in large foundation models*. arXiv preprint arXiv:2309.05922. URL: https://doi.org/10.48550/arXiv.2309.05922

[37] Y. Zhang, Y. Li, L. Cui, D. Cai, L. Liu, T. Fu, X. Huang, E. Zhao, Y. Zhang, Y. Chen, L. Wang, A.T. Luu, W. Bi, F. Shi, and S. Shi, (2023). *Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models*. arXiv preprint arXiv:2309.01219. URL: https://doi.org/10.48550/arXiv.2309.01219.

[38] Ollama. *Get up and running with large language models*. URL: https://ollama.com/. Accessed 17.05.2024.

[39] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al., (2023). *Llama 2: Open*

Foundation and Fine-Tuned Chat Models. arXiv preprint arXiv:2307.09288. URL: https://doi.org/10.48550/arXiv.2307.09288.

[40] Ollama. *Ollama*. GitHub. URL: https://github.com/ollama/ollama. Accessed 17.05.2024.

[41] Ollama. *Models*. URL: https://ollama.com/library. Accessed 17.05.2024.

[42] Meta. *Introducing Meta Llama 3: The most capable openly available LLM to date*. URL: https://ai.meta.com/blog/meta-llama-3/. Accessed 17.05.2024.

[43] A.Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D.S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al., (2023). *Mistral 7B*. arXiv preprint arXiv:2310.06825. URL: https://doi.org/10.48550/arXiv.2310.06825.

[44] OpenAI. *Introducing ChatGPT*. URL: https://openai.com/index/chatgpt/. Accessed 17.05.2024.

[45] J. Deng, and Y. Lin, (2023). *The Benefits and Challenges of ChatGPT: An Overview*. Frontiers in Computing and Intelligent Systems, Vol. 2, No. 2, pp. 81-83. URL: https://doi.org/10.54097/fcis.v2i2.4465.

[46] B. Chen, Z. Zhang, N. Langrené, and S. Zhu, (2023). *Unleashing the potential of prompt engineering in Large Language Models: a comprehensive review*. arXiv preprint arXiv:2310.14735. URL: https://doi.org/10.48550/arXiv.2310.14735.

[47] P. Sahoo, A.K. Singh, S. Saha, V. Jain, S. Mondal, A. Chadha, (2024). *A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications*. arXiv preprint arXiv:2402.07927. URL: https://doi.org/10.48550/arXiv.2402.07927.

[48] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q.V. Le, and D. Zhou, (2022). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. In: Advances in Neural Information Processing Systems, Vol. 35, pp. 24824–24837. URL: https://doi.org/10.48550/arXiv.2201.11903.

[49] A. Roberts, C. Raffel, and N. Shazeer, (2020). *How Much Knowledge Can You Pack Into the Parameters of a Language Model?* arXiv preprint arXiv:2002.08910. URL: https://doi.org/10.48550/arXiv.2002.08910.

[50] OpenAI. *Pricing*. URL: https://openai.com/api/pricing/. Accessed 18.05.2024.

[51] X Developer Platform. *X API*. URL: https://developer.x.com/en/products/twitter-api. Accessed 18.05.2024.

[52] J. Chaker, and O. Habib, (2007). *Genre categorization of web pages*. In Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007), pp. 455-464. IEEE. URL: https://doi.org/10.1109/ICDMW.2007.120.

[53] S. Meyer zu Eissen, and B. Stein, (2004). *Genre Classification of Web Pages*. In KI 2004: Advances in Artificial Intelligence, pp. 256-269. URL: https://doi.org/10.1007/978-3-540-30221-6_20.

[54] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery, (1998). *Learning to extract symbolic knowledge from the word wide web*. In proceeding of the 15th national/10th conference on artificial intelligence/innovative applications of artificial intelligence, Madison, W.

[55] E. Baykan, M. Henzinger, L. Marian, and I. Weber, (2011). *A Comprehensive Study of Features and Algorithms for URL-Based Topic Classification*. ACM Transactions on the Web, Vol. 5, Issue 3, Article 15 (July 2011), pp. 1-29. URL: https://doi.org/10.1145/1993053.1993057.

[56] Curlie. *Collect the best websites for any topic!* URL: https://curlie.org/. Accessed 20.05.2024.

[57] Internet Archive. *The Wayback Machine*. URL: https://wayback-api.archive.org/. Accessed 21.05.2024.

[58] Internet Archive. *Internet Archive's Terms of Use, Privacy Policy, and Copyright Policy*. URL: https://archive.org/about/terms.php. Accessed 21.05.2024.