University of Verona
Department of Computer Science
Master's degree in Data Science

Enhancing Query Recommendations Through User Behavior Analysis

Supervisor:
Prof. Elisa Quintarelli

Co-Supervisor:
Dott. Niccolò Marastoni

Candidate:
Federico Vaona
VR 495585

ACADEMIC YEAR 2024/2025

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities in natural language generation; however, their outputs often remain generic when lacking personalization. This thesis examines whether lightweight, interpretable user profiles can be effectively integrated into LLM-based query suggestions. Building on the K-LaMP framework, a recent approach for knowledge-augmented query suggestion, we extend its entity-centric knowledge store with descriptors from ORCID profiles and with synthetic attributes such as nationality, profession, and hobbies. The experimental setting focuses on cultural tourism in Verona, where user interactions with Points of Interest (POIs) are combined with profile information to generate recommendations for the next query. Evaluation yields three main findings: (i) entity-centric personalization enhances relatedness and usefulness of suggestions compared to contextual baselines; (ii) ORCID keywords provide domain-specific grounding, while synthetic attributes broaden personalization; (iii) prompt design critically determines how effectively user signals influence LLM outputs. Overall, the findings demonstrate that combining contextual information with structured user profiles enables richer and more relevant query recommendations, offering a scalable path for real-world deployment in domains such as academic search, tourism, and e-commerce.

Contents

Chapter 1

Introduction

1.1 Context and Motivation

The last two decades have witnessed an exponential growth in the number of Internet users, which has made search engines indispensable tools for satisfying information needs on the web.

An early study showed that the length of the user's query is generally very short [1]. Due to their short size, most of the time, these queries were ill-formed, vague, and ambiguous. Furthermore, almost 78.2% of users modify their original queries to those recommended by the search engine if they are not satisfied with the results obtained using their original queries. In this context, it is essential for search engines to produce high-quality recommendations that enable users to access the desired and relevant information more quickly.

From Query Logs to LLMs "Related Searches" is a fundamental module of a Search Engine Result Page (SERP), typically powered by query recommendation systems. Traditional systems relied heavily on query logs, analysing how large numbers of users reformulated their queries. For example, Feuer et al. [2] analysed more than 1.5 million queries from a commercial search engine and found that query suggestions accounted for roughly 30% of all queries, highlighting their central role. Other studies confirmed that query suggestions are especially valuable for difficult or unfamiliar topics [3], although these systems suffer from the well-known *cold-start problem*[1] [4].

Early attempts to move beyond logs (e.g., Bhatia et al. [5]) were limited by a lack of semantic understanding: models could only count words and

[1]The *cold-start problem* occurs when a system cannot provide accurate recommendations due to the lack of sufficient data on new users or new items.

phrases, resulting in semantic errors and poor fluency. In contrast, modern Large Language Models (LLMs) such as GPT-4 and Gemini utilize deep learning to construct contextual and semantic representations, generating novel, coherent text [6, 7]. This makes them particularly attractive for query recommendation, as they can handle ambiguity, generate fluent alternatives, and generalize across domains.

Challenges of Personalization Despite these successes, the goal of making LLMs more meaningful and relevant for every user remains an open challenge. Even if they generate high-quality text in general contexts, the main problem emerges when tailoring outputs to an individual user requires accounting for variables such as preferences, expertise, or long-term knowledge. Naïvely prompting an LLM with user history is impractical due to context-window limits and scalability issues. Moreover, personalization through fine-tuning is prohibitively costly for individual users [8]. Recent studies have explored personalization via in-context learning (ICL) [9, 10, 11, 12], where user descriptors are inserted directly into the prompt. This approach shows promise—handling cold-starts, improving semantic richness, and offering interpretability—but remains highly sensitive to prompt design, limited by history length, and computationally demanding.

Other research has attempted to incorporate deeper user profiles [13, 14], sometimes at the cost of privacy and scalability. Thus, while LLMs already perform well in domains like query suggestion or conversational recommendation, the lack of robust, user-aware personalization remains a key limitation [15, 16].

1.2 Contributions of this Thesis

The objective of this thesis is to reimplement and extend one of the most recent frameworks for next-query recommendation, K-LaMP [8], with the specific goal of enhancing personalization through user behaviour analysis. Whereas the original K-LaMP demonstrated the effectiveness of entity-centric knowledge in guiding LLMs, this work investigates whether additional, lightweight user profiles can further improve the quality and stability of generated suggestions.

The main contributions are as follows:

- **Reproducible reimplementation:** a reproducible version of K-LaMP is recreated using *Points of Interest (POIs)* from Verona—such as museums, monuments, and landmarks—as simulated browsing content and synthetic sessions (see Chapter 4 for details).

- **Profile enrichment:** user modelling is extended by incorporating structured academic descriptors (ORCID keywords) and synthetic persona attributes (e.g., nationality, profession, hobbies) to complement behavioural signals.

- **Prompt engineering:** alternative prompt formulations are tested to explicitly guide LLMs in prioritizing user-specific information, with the aim of improving personalization without sacrificing novelty or diversity.

Overall, the thesis presents a reproducible framework for studying entity-centric personalization in LLM-based query recommendation and examines how user-aware profiles and carefully designed prompts can enhance the effectiveness, interpretability, and user relevance of recommendations.

1.3 Structure of the Thesis

The thesis is structured as follows.

- **Chapter 2** introduces the key background concepts, including recommendation systems, query suggestion, information retrieval, natural language processing, entities, knowledge graphs, large language models, and prompt engineering. These foundations establish the terminology and tools for the subsequent analysis.
- **Chapter 3** reviews the most relevant contributions in the literature, with particular emphasis on the K-LaMP framework and related approaches to personalized query recommendation. Each study is discussed in terms of objectives, methods, findings, and relevance to this work.
- **Chapter 4** presents the methodological contribution of the thesis. It describes the reimplementation and extension of the K-LaMP pipeline under a reproducible setting, including ORCID-based profiles, synthetic attributes, and prompt design strategies.
- **Chapter 5** reports the experimental setup and results. It compares the three proposed models and evaluates them through a combination of examples and qualitative analysis.
- **Chapter 6** concludes the thesis by summarizing the main findings, discussing limitations, and outlining possible directions for future work, including extensions toward real-world deployment and multi-modal personalization.

Chapter 2

Background

This chapter introduces the key background concepts that underpin the rest of this thesis. The goal is not to provide a comprehensive survey of the entire field, but rather to outline the notions most relevant for understanding the proposed framework. Each section focuses on a foundational area, ranging from recommendation systems and query suggestion to information retrieval, natural language processing, entities, and large language models. Simple examples are provided where appropriate to clarify the concepts. Throughout the thesis, we refer to *Points of Interest (POIs)*—such as monuments, museums, and landmarks in Verona—as the textual "documents" used to simulate browsing and clicks; we also use *ORCID* (Open Researcher and Contributor ID) keywords as structured descriptors of a user's academic profile. Details are provided in Chapter 4.

2.1 Recommendation Systems

Recommendation systems are a broad class of algorithms designed to assist users in navigating large information spaces by automatically suggesting items of potential interest. They are now ubiquitous in digital platforms: Amazon recommends products, Netflix and Spotify suggest movies or songs, and academic search engines such as Google Scholar or Semantic Scholar highlight papers related to a user's research interests.

The key idea is to leverage information about user preferences, past interactions, or item characteristics to predict what the user is most likely to find useful next. Classical works in the field [17, 18] distinguish between two main paradigms:

- **Content-based filtering.** Items are recommended based on their similarity to those already consumed or liked by the user. For example, if a user frequently watches historical documentaries, the system may recommend other documentaries tagged with "history" or "culture." Features used for similarity can include keywords, categories, or structured metadata. The main advantage of this approach is personalization, but it suffers from the so-called *overspecialization problem*, as users may only be exposed to items that are too similar to their past choices.

- **Collaborative filtering.** Recommendations are generated from patterns in the behaviour of many users. If user A and user B have similar viewing histories, items watched by A but not yet by B can be recommended to B. Collaborative filtering methods include neighbourhood-based algorithms (user-based and item-based kNN) and matrix factorization techniques [19]. This approach captures complex associations beyond explicit item features but is vulnerable to the *cold-start problem*, where new users or new items lack sufficient data.

To overcome the limitations of pure approaches, **hybrid systems** combine both paradigms, often augmented with contextual information such as time, location, or demographics [20]. More recently, advances in **deep learning** have introduced neural recommendation models that exploit embeddings of users and items, while **Large Language Models** (LLMs) have enabled recommendation to be reframed as a language generation problem [9, 10]. This opens opportunities for richer personalization, allowing recommendations to be expressed in natural language and adapted to user profiles and context.

In this thesis, query recommendation is framed as a specialized form of recommendation system, where the "items" are not products or movies but new search queries. The goal is analogous: to anticipate what information the user might want to explore next, thereby improving efficiency, discovery, and personalization in information retrieval.

2.2 Query Recommendation

Query recommendation aims to help users refine or extend their search by suggesting alternative or follow-up queries. For example, after searching for *"Arena di Verona concerts"*, a system might recommend *"opera season in Verona"* or *"Roman amphitheatres in Italy"*. Such suggestions enhance both efficiency (by helping users find what they want faster) and discovery (by introducing new, yet relevant, aspects).

Traditional query recommendations were generated from query logs, analysing how large numbers of users reformulated their queries. More recently, neural models and Large Language Models (LLMs) have been used to generate semantically richer suggestions that go beyond simple reformulations. Evaluation typically involves metrics such as *precision@k* or human judgments of *relatedness*, *usefulness*, and *novelty*.

2.3 Information Retrieval Basics

Information Retrieval (IR) is the process of retrieving relevant information from large collections of documents based on a user query. A typical IR system, such as a web search engine, receives a query like *"historic buildings in Verona"* and returns a ranked list of documents (e.g., Wikipedia pages, travel websites, or POI descriptions).

Classical IR relied on keyword-based models, such as TF-IDF or BM25, which score documents based on term frequency and document rarity. These methods are effective for exact word matches but are limited when the query and the relevant documents use different vocabulary. Modern IR integrates semantic representations (e.g., word embeddings, transformer-based encoders) that capture meaning beyond keywords.

2.4 Natural Language Processing

Natural Language Processing (NLP) is a branch of artificial intelligence that focuses on enabling computers to process and generate human language. It combines methods from linguistics, computer science, and machine learning to transform unstructured text into structured information.

Typical NLP tasks include:

- **Tokenization:** splitting text into units such as words or sentences.
- **Part-of-Speech (POS) tagging:** assigning grammatical categories (e.g., noun, verb, adjective) to each token.
- **Named Entity Recognition (NER):** detecting and classifying mentions of real-world entities such as people, places, and organizations.
- **Dependency parsing:** analysing the syntactic structure of a sentence by identifying relationships between words.
- **Text classification:** assigning labels to documents or sentences, for example, detecting topics or sentiment.

In recent years, NLP has been revolutionized by the adoption of deep learning models, from recurrent neural networks to transformer-based architectures. These methods allow systems to capture complex semantic and syntactic patterns, paving the way for applications such as machine translation, chatbots, question answering, and query recommendation. In this thesis, NLP techniques are mainly used for **entity extraction**, which involves identifying mentions of concepts such as *Arena di Verona*, *Italy*, or *Roman amphitheatre* in user queries or document text. The spaCy library provides pre-trained models for this task, making it possible to automatically detect and categorize entities that are then stored in the personalized knowledge store introduced in Chapter 3 and 4.

2.5 Entities

An *entity* is a real-world object, concept, or place that can be explicitly identified within a query or a document. Examples include *Arena di Verona*, *Roman amphitheatre*, or *Italy*. Entity extraction is the process of detecting such terms in text, using natural language processing (NLP) techniques.

Entities are central to query recommendation because they provide a semantic representation of what the user is interested in, going beyond the surface form of the query. For example, the queries *"concerts in the Arena"* and *"opera season Verona"* may look different lexically, but they both contain the entity *Arena di Verona*. By grounding personalization on entities, a system can connect queries that are semantically related even when they share no keywords.

In this thesis, entities were extracted using **spaCy**, a widely adopted open-source NLP library that provides pre-trained statistical and transformer-based models for Named Entity Recognition (NER). For instance, when processing the query *"visit Arena di Verona in summer"*, spaCy outputs:

`[Arena di Verona (FAC[1]), Verona (GPE[2]), summer (DATE[3])]`

where the entity spans are identified, along with their corresponding semantic labels.

The extracted entities are then aggregated into a **personal knowledge store**, which records their frequency and recency for each user. This compact representation forms the basis for the personalization strategies evaluated in this work.

[1]FAC = Facility, i.e., a human-made structure such as a building, airport, or monument.
[2]GPE = Geopolitical Entity, e.g., countries, cities, states, or regions.
[3]DATE = Temporal expressions, e.g., years, months, or generic periods like "summer".

2.6 Knowledge Graphs

A *Knowledge Graph* (KG) is a structured representation where entities are nodes and relations between them are edges. For example, in a cultural heritage KG, the node *Arena di Verona* could be linked to *Verona* (location), *Roman amphitheatre* (type), and *Opera Festival* (event). This enables navigation not only across documents, but also across semantic relationships.

Knowledge graphs have been widely used in query recommendation and information retrieval. They enable systems to suggest related entities or paths: if a user queries *Arena di Verona*, the system may recommend *Roman theatres in northern Italy* or *Castelvecchio Museum*, leveraging graph connectivity.

However, building and maintaining large KGs (e.g., DBpedia, Wikidata) poses scalability and heterogeneity challenges. In this thesis, we do not construct a full KG; instead, we adopt an entity-centric approach inspired by KG-based reasoning, where entities are extracted and aggregated into a personalized store. This strikes a balance between semantic richness and practical scalability.

2.7 Large Language Models

Large Language Models (LLMs) are neural networks trained on massive corpora of text to predict and generate natural language. Well-known examples include GPT-3 [21], GPT-4 [6], and more recently Gemini [7]. Their strength lies in their ability to generalize: even without task-specific training, they can produce coherent outputs for tasks such as summarization, translation, or query recommendation.

For instance, when presented with the query *"Roman amphitheatres in northern Italy"*, an LLM can suggest related queries such as *"Arena di Verona opening hours"* or *"Colosseum vs. Arena di Verona architecture"*. Such outputs illustrate the model's ability to link historical landmarks and provide semantically coherent continuations that go beyond simple keyword matching.

This capability makes LLMs powerful engines for producing diverse and meaningful query recommendations. At the same time, they are highly sensitive to input phrasing (prompt design) and limited by the maximum context window, which constrains how much user history can be incorporated. These challenges motivate the integration of additional mechanisms—such as entity-based knowledge stores and profile enrichment—explored in this thesis.

2.8 Prompt Engineering

Prompt engineering refers to the practice of designing the input given to a Large Language Model (LLM) so that the generated output better aligns with the user's goals. Because LLMs are highly sensitive to how instructions are phrased, even small changes in wording can lead to very different results [22].

For example, asking an LLM *"Suggest another tourist attraction in Verona"* might yield generic suggestions such as *"Piazza delle Erbe"*, while a more specific instruction like *"As a history professor interested in Roman architecture, what should I visit in Verona?"* directs the model toward contextually richer answers such as *"Arena di Verona"* or *"Porta Borsari"*.

In this thesis, prompt engineering is employed to investigate how various formulations—ranging from simple additions of user descriptors to explicit guidance on their importance—influence the personalization and stability of query recommendations.

2.9 User Profiling

User profiling refers to the process of building a structured representation of a user's interests, preferences, or expertise to personalize recommendations and retrieval.

Profiles can be constructed from different types of signals:

- **Behavioural:** derived from past interactions such as queries, clicks, and viewed documents. These logs capture user behaviour but raise privacy concerns and may suffer from sparsity.
- **Demographic:** attributes such as age, nationality, or profession, often collected explicitly. These can support cold-start scenarios but may be too coarse.
- **Academic:** structured descriptors like author keywords or domain-specific terms, which summarize expertise and areas of interest in a standardized way.

User profiling is central to personalization: it allows a system to tailor its suggestions beyond generic popularity signals. At the same time, it introduces challenges such as protecting sensitive information, avoiding stereotyping, and handling evolving interests.

Classic studies in personalization for dialogue and recommendation (e.g., Zhang et al. [14], Komeili et al. [13]) demonstrated how explicitly modelling user attributes can lead to richer interactions.

An important distinction is between **short-term personalization**, which adapts to the user's immediate session context (recent queries and clicked documents), and **long-term personalization**, which relies on more stable descriptors such as demographic attributes or declared expertise. While the former captures the evolving intent within a search session, the latter provides continuity across sessions and supports deeper personalization.

2.10 Session-Based Search

Traditional information retrieval often treats each query as an isolated event. In practice, however, users typically issue a **sequence of related queries** within a single search session. For example, a visitor might first search for *"Arena di Verona concerts"*, then refine the query to *"opera season 2024 Verona"*, and finally explore *"tickets for Arena di Verona opera festival"*.

Session-based search recognizes that queries are not independent, but linked by a user's evolving intent. This has two main implications:

- Systems should model dependencies across queries, not just optimize for single-query relevance.
- Recommendations should anticipate likely next steps, helping the user continue their exploration efficiently.

Earlier neural models, such as the hierarchical encoder-decoder by Sordoni et al. [23] and attention-based extensions by Dehghani et al. [24], explicitly addressed session continuity, while more recent work combines session-level and long-term signals [25].

Query recommendation is thus closely tied to session modelling: suggesting the next query requires understanding both the **short-term context** (current session) and the **long-term profile** (user history). This motivates the entity-centric approach adopted in this thesis, where past entities and profile descriptors guide next-query generation.

Chapter 3

Related Work

This chapter reviews the most relevant contributions to query recommendation, with a particular focus on works that inform and inspire the present thesis. Each subsection focuses on a representative study, outlining its objectives, methodologies, findings, and limitations, while situating it within the broader literature and clarifying its specific relevance for this work. Where relevant, we briefly relate each work to our experimental setting, in which *Points of Interest (POIs)* from Verona are used to model the browsed content and *ORCID* keywords provide a compact representation of users' academic interests (see Chapter 4).

3.1 K-LaMP: Knowledge-Augmented LLMs (Baek et al., 2024)

The work of Baek et al. [8] represents the most direct inspiration for this thesis. K-LaMP (*Knowledge-Augmented Large Language Models for Personalized Contextual Query Suggestion*) was introduced as a framework to enhance query suggestion by combining the generative capabilities of LLMs with lightweight, entity-centric personalization.

Motivation and Context

Prior approaches to query recommendation faced multiple limitations. Log-based and statistical methods were effective on frequent queries but lacked semantic generalization. Neural and context-aware models improved contextual predictions but were still constrained to session-level signals. Even with the rise of LLMs, personalization remained shallow: directly injecting user history into prompts was limited by context-window size and the in-efficiency of handling long interaction histories. K-LaMP was proposed to address these challenges by introducing a scalable, privacy-conscious, and semantically meaningful representation of user knowledge.

Framework Design

The K-LaMP framework is built around two main components:

- **Memory Stream:** a chronological record of user interactions, including issued queries and clicked documents, enriched with timestamps and minimal metadata.
- **Entity Knowledge Store:** a dynamic store that aggregates entities extracted from the memory stream. Each entity is associated with occurrence counts and recency information, providing a compact representation of what the user "knows" or has shown interest in.

Entities are retrieved from the store using three strategies:

1. *Familiar:* entities frequently and recently interacted with.
2. *Unfamiliar:* entities related to the user's context but less frequently seen.
3. *Lapsed:* entities from older interactions, reintroduced after inactivity.

These strategies enable various forms of personalization, striking a balance between exploiting known preferences and exploring new or previously overlooked topics.

Integration with LLMs

The retrieved entities are combined with the user's current query, session context, and the content of the currently viewed document. This structured prompt is fed into an LLM (e.g., GPT-3.5, as used in the original experiments), which then generates a next-query suggestion. The design leverages the LLM's generative flexibility while grounding it in the user's entity-centric profile. Unlike full-history prompts, the entity store is compact, interpretable, and scalable across millions of users.

Evaluation and Findings

K-LaMP was evaluated using large-scale Bing search logs, which provided realistic user sessions. The study relied on human evaluation across three dimensions:

- **Relatedness:** whether the suggested query is topically connected to the user's current context.
- **Usefulness:** whether the suggestion is practically helpful for continuing the search.
- **Novelty:** whether the suggestion introduces new and interesting aspects without being redundant.

The results showed that K-LaMP significantly outperformed both context-only baselines and LLM prompts without knowledge augmentation. In particular, entity-centric personalization resulted in more useful and relevant suggestions while maintaining novelty.

Limitations

Despite its success, K-LaMP has limitations. The reliance on proprietary Bing logs limits reproducibility for independent researchers. The quality of personalization is strongly dependent on accurate entity extraction. Finally, while entity stores are compact, their integration into prompts still requires careful design to avoid overwhelming the LLM.

Relevance for This Thesis

K-LaMP forms the foundation of this work. Our thesis extends the framework in three ways:

- By recreating the pipeline in a **synthetic, reproducible setting**, using POIs from Verona as documents and simulated user sessions.
- By enriching user profiles with **ORCID keywords**, introducing structured academic descriptors beyond behavioural logs.
- By exploring **prompt engineering strategies**, explicitly guiding the LLM to prioritize certain user signals (ORCID keywords, persona attributes) and testing how different prompt formulations affect personalization.

Through these extensions, the thesis aims to validate whether entity-centric personalization remains effective under realistic data constraints and whether structured profiles can further improve personalization in query recommendation.

3.2 Graph-Query Suggestions for Knowledge Graph Exploration (Lissandrini et al., 2020)

Lissandrini et al. [26] proposed query suggestions for knowledge graph (KG) exploration. Their method leverages query logs, along with graph topology, to guide users toward semantically meaningful entities and paths. The contribution lies in combining log-based usage patterns with KG structure, bridging statistical query suggestion and structured navigation. The study evaluated KG exploration tasks on DBpedia, a large-scale knowledge base automatically extracted from Wikipedia, and found that graph-aware suggestions improved both the efficiency and coverage of user exploration.

Similar graph-based strategies had already been studied in random walks on click graphs [27] and semantic query reformulation [28], but this work was among the first to systematically tailor query suggestions to knowledge graph settings. A key limitation is scalability, as graphs can be extremely large and heterogeneous.

Relevance for this thesis: This paper demonstrates how structured representations (entities in a KG) can improve personalization. It supports the idea behind K-LaMP and our extension: grounding suggestions in entity-centric representations, specifically those extracted from POIs and ORCID profiles.

3.3 Boomerang: Proactive Insight-Based Recommendations (Lee et al., 2021)

Lee et al. [29] introduced *Boomerang*, a system that provides proactive recommendations during data exploration. Unlike traditional reactive query suggestion, Boomerang anticipates potential information needs by monitoring user interactions and surfacing relevant queries and insights ahead of time. The method was validated in the SIGMOD 2021 paper[1] through experiments with interactive datasets, showing improvements in the discovery of useful information.

This proactive paradigm resonates with earlier work on exploratory search [30] and interactive IR systems [31]. Its limitation lies in balancing proactivity with user control, avoiding information overload or irrelevant suggestions.

[1] ACM SIGMOD (Special Interest Group on Management of Data) is one of the leading international conferences on data management.

Relevance for this thesis: Boomerang highlights the importance of moving beyond passive, log-based recommendations. Our work shares this vision: query suggestions should adapt proactively to user context and profiles, not just repeat past behaviours.

3.4 Generating Query Recommendations via LLMs (Bacciu et al., 2024)

Bacciu et al. [32] presented one of the first systematic studies on using Large Language Models (LLMs) for query recommendation. Their goal was to test whether generative models, which have shown strong generalization ability in tasks such as GPT-3 and GPT-4, could replace or complement traditional log-based and neural approaches. The authors compared zero-shot and few-shot prompting strategies against baselines, including statistical models and deep neural networks. Evaluation combined standard information retrieval metrics with human judgments of relatedness and usefulness. Results demonstrated that LLMs can produce competitive and often more semantically diverse query suggestions. At the same time, the study highlighted that quality was highly sensitive to the prompt design and the amount of contextual information included; even small variations in prompt wording could significantly affect the generated queries.

This work fits within a broader research direction that frames recommendations as a language generation task. For instance, Geng et al. [9] introduced P5, a paradigm where various recommendation tasks are reformulated as natural language prompts; Ji et al. [10] proposed GenRec, leveraging generative models conditioned on user histories; and Lyu et al. [12] investigated prompting strategies to adapt recommendations to different user contexts. Taken together, these contributions indicate that LLMs are promising engines for generative personalization, though their effectiveness depends heavily on careful input design and contextual guidance.

Despite these promising results, several limitations remain. Experiments often rely on commercial APIs, which raises issues of reproducibility and cost compared to open neural models. Furthermore, the limited context window of current LLMs restricts the amount of user history that can be incorporated, making it difficult to achieve deep personalization without external memory structures. These findings make it clear that, while LLMs open up exciting opportunities for query recommendation, their integration requires additional mechanisms to ensure stability, scalability, and personalization.

Relevance for this thesis: The study confirms that LLMs can generate diverse and relevant query suggestions, but also highlights their fragility in handling user history and prompt sensitivity. Building on these insights, this thesis extends the K-LaMP framework by integrating structured user profiles (via ORCID) and refining prompt design to achieve more stable and user-aware personalization.

3.5 Support the Data Enthusiast (Morton et al., 2014)

Morton et al. [33] argued for systems designed to assist "data enthusiasts"—users without advanced technical expertise. They proposed integrating visualization, enrichment, and recommendation within a single framework, lowering barriers to entry for exploratory data analysis.

This vision resonates with modern human-in-the-loop systems [34] and context-aware data recommendations [35]. While not limited to query recommendation, it influenced how assistive systems can incorporate user context. The limitation is that it remained more conceptual, with limited large-scale empirical validation.

Relevance for this thesis: This paper emphasizes usability and non-expert accessibility. In our tourism-based experiments, we similarly aim to build a framework that remains interpretable and usable without requiring large-scale proprietary logs.

3.6 Optimizing Data Delivery (Luera et al., 2025)

Luera et al. [36] focused on personalization from the angle of *data delivery*. Instead of only predicting content, their system adapts to user preferences over modality (visuals, tables, text). Through user studies, it was shown that tailoring the delivery format significantly affects perceived usefulness and satisfaction.

This expands the scope of personalization beyond "what" is recommended to also "how" it is presented. Related perspectives can be found in multimodal IR [37] and adaptive visualization systems [38]. Its limitation lies in generalizing presentation preferences across diverse users.

Relevance for this thesis: While our work focuses on what queries are generated, this study points to an additional direction for future extensions: tailoring not only the content of recommendations but also their presentation. Depending on user characteristics and profiles, future systems could adapt whether next queries are delivered as plain text, structured lists, graphs, or even images.

3.7 A Cooperative Co-Evolutionary Genetic Algorithm (Barman et al., 2024)

Barman et al. [39] proposed *QRMOCCGA*, a cooperative co-evolutionary genetic algorithm for query recommendation. The method formulates query suggestion as a multi-objective optimization problem, optimizing simultaneously for *relevance* and *diversity*. By simulating an evolutionary process—where candidate queries are iteratively combined, mutated, and selected—the algorithm produces recommendations tha, such as association-rule mining [40] and utility-based ranking [41], but extends them with an evolutionary framework capable of handling multiple objectives simultaneously-based ranking [41], but extends them with an evolutionary framework capable of handling multiple objectives at once. The main limitation is the **computational cost**, which makes real-time deployment in large-scale systems challenging.

Relevance for this thesis: Although we do not use evolutionary algorithms, this work emphasizes the importance of balancing *relevance* and *diversity*. In our evaluation, we test whether ORCID-based personalization can improve the usefulness of generated query suggestions without compromising novelty.

Summary

The reviewed literature highlights a variety of strategies for query recommendation, ranging from log-based analysis and graph-based exploration to proactive systems, generative LLMs, and optimization-driven algorithms. Each approach has contributed valuable insights, while also exposing open challenges in scalability, depth of personalization, and usability. Among these, K-LaMP emerges as a pivotal framework, demonstrating how entity-centric knowledge can anchor LLM-based personalization. Building on this foundation, this thesis explores whether integrating structured user profiles (via ORCID) and refined prompt engineering can further enhance the quality and interpretability of personalized query recommendations.

Chapter 4

Personalized Query Recommendations

4.1 Overview of the Thesis Contribution

This chapter details the methodological contribution of the thesis, focusing on the design, implementation, and evaluation of three progressively enriched versions of the K-LaMP framework [8] for personalized contextual query suggestion. The original K-LaMP demonstrated that augmenting Large Language Models (LLMs) with user knowledge enhances the relevance of query suggestions, but it relied on large-scale Bing search logs that are not publicly available. This limitation is consistent with prior work noting the restricted availability of large-scale commercial logs for academic research [35, 25].

In this work, user sessions are instead **simulated through Points of Interest (POIs) in Verona**: POI descriptions are used as page content, and synthetic query histories are constructed around them to emulate user behavior. The study further investigates how alternative strategies for presenting such information to the LLM, particularly through prompt design, influence the degree of personalization achieved.

Specifically, we extend the framework in three directions:

1. **ORCID keywords as auxiliary entities:** keywords extracted from ORCID profiles are added to the context to complement simulated sessions and personalize the next query recommendation.

2. **Prompt engineering:** we modify the instructions given to Gemini so that personal entities (starting with ORCID keywords) are explicitly prioritized during generation.

3. **Synthetic enriched profiles:** we introduce additional attributes (nationality, profession, hobbies) into user profiles, and adapt the prompt so that these features carry more weight than traditional academic keywords.

The goal is not only to replicate the K-LaMP pipeline under a reproducible and privacy-preserving setting, but also to analyse how personalization evolves under increasing levels of user information and prompt guidance.

4.2 Data Simulation and Construction of the Entity-Based Knowledge Store

Unlike the original study, we did not have access to real-world user logs. Realistic behavioural data were therefore **synthetically generated** to approximate search interactions while avoiding privacy and accessibility issues. The dataset of Points of Interest (POIs) in Verona served as the foundation for this process.

The generation procedure followed these steps:

- For each POI, its textual description was treated as if it were the web page clicked and viewed by the user.
- Queries were constructed to represent the user's interaction with the POI, forming a synthetic session history.
- From both the query text and the POI description, entities were extracted at each step of the session, and their occurrences were updated in the entity-based knowledge store.

4.3 Entity Retrieval Strategy

In the original K-LaMP framework, user entities could be retrieved according to three different strategies [8]:

- *familiar*
- *unfamiliar*
- *lapsed*

For example, in a tourism scenario, a visitor might frequently search for the *Piazza Bra* (familiar), occasionally for the *Castelvecchio Museum* (unfamiliar), and not have interacted with the *Verona Airport* for a long time (lapsed). These categories were meaningful in the context of long, real-world user histories as studied in session-based recommendation [23, 24]. In our case, since the histories are synthetically generated and relatively short, the distinction among these strategies produced little variation. For this reason, we restricted our implementation to the **familiar** strategy, which emphasizes entities most frequently associated with the user. This simplification ensured that retrieval remained interpretable, while allowing us to focus on the role of prompt design and profile richness.

Example of Knowledge Store Update. After visiting the POI *"Arena di Verona"* and issuing a query such as *"Roman amphitheaters in northern Italy"*, the following entities may be extracted:

[*Arena di Verona*, *Roman amphitheater*, *Italy*]

If these entities already appeared in past interactions of the same user, their frequency counts are incremented in the knowledge store. Otherwise, they are added with an initial occurrence value. This process produces a dynamic, entity-centric memory of the user's search history.

4.4 Example of Next Query Suggestion Generation Process

Consider the case where user u_1 issues the query *"historic gelaterias in Verona"* and opens the page of the POI *"Gelateria Savoia"*. From the query text and the POI description, the following entities are extracted:

[*Gelateria Savoia*, *Piazza Bra*, *Verona*]

The personal knowledge store of u_1 shows that *Piazza Bra* and *Verona* have appeared frequently in past interactions. According to the *familiar* strategy, these entities are therefore prioritized and selected as contextual signals. They are passed to Gemini together with the broader session context (i.e., the current query, the selected page, and the session history), which the model then uses to generate the next personalized query suggestion.

4.5 Algorithmic Implementation (Conceptual Pipeline)

The reimplementation follows a lightweight, six-step pipeline. We present it in conceptual pseudocode to emphasize the main phases rather than implementation details.

Algorithm 1 Conceptual K-LaMP Pipeline

Require: u (user), q_t (current query), w_t (current article text), $profile$ (ORCID keywords and/or synthetic attributes)

```
procedure NextQuery(u, q_t, w_t, profile)
    add_to_memory_stream(u, q_t, w_t)
    K ← build_entity_store(u)
    P ← extract_personal_entities(K, strategy = familiar)
    S ← user_query_session(u)
    prompt ← build_prompt(q_t, S, w_t, P, profile, variant)
    q̂_{t+1} ← generate_next_query(Gemini, prompt)
    return q̂_{t+1}
```

Step descriptions.

- **add_to_memory_stream**: appends the current interaction to the user's memory stream (query q_t and clicked/seen article w_t with timestamps and simple metadata). This forms the chronological log used by subsequent steps.
- **build_entity_store**: parses the user's memory stream (past queries, viewed pages, and ORCID keywords); normalizes entities and aggregates them into a frequency–recency summary per user. Returns K, the *Entity Knowledge Store*.
- **extract_personal_entities**: selects a subset of entities from K to expose to the LLM. In this thesis, we use the *familiar* strategy (top frequent and recent entities) given the synthetic, short histories.
- **user_query_session**: retrieves a short, ordered window of the user's recent queries (the session context), used to preserve intent continuity and avoid lexical repetition.
- **build_prompt**: assembles the LLM input from the current query, session, current article, personal entities, and (when available) ORCID keywords and/or synthetic persona attributes. The wording (guidance/weights) differs across our three variants.
- **generate_next_query**: calls Gemini with the constructed prompt and post-processes the output to return exactly *one* concise next-query suggestion.

4.6 Model 1: ORCID Keywords

The first variant is designed to stay as close as possible to the original K-LaMP design. The input to Gemini includes:

- The current query,
- The session history,
- The page content (POI description),
- A set of **personal entities**, extracted from the user's knowledge store using the *familiar* retrieval strategy,
- A set of **ORCID keywords**, directly taken from a user profile and accompanied by their definitions.

In this model, the ORCID keywords are injected as additional signals on top of the standard K-LaMP context, which already includes personal entities derived from user interactions. The aim is to enrich the user model with academic descriptors and explore whether such metadata can contribute to more personalized query suggestions, a topic already considered in earlier work on user profiling for personalization [14, 13].

4.7 Model 2: Prompt-Enhanced ORCID

The second variant addresses the limitation observed in Model 1. Instead of only appending ORCID keywords to the context, we redesigned the prompt so that Gemini would give them explicit priority. This builds on evidence that LLMs are highly sensitive to prompt design and benefit from explicit guidance [12]. In simple terms, the model was instructed to *"pay special attention to the ORCID keywords when generating the next query."* The intuition is that by making the importance of these keywords explicit, they would be more consistently incorporated into the generated suggestions.

Prompt template used in Model 2. For transparency and reproducibility, the exact instructions used in our experiments were the following:

```
Guidance and priorities:
- Prioritize long-term user relevance from ORCID keywords.
- Maintain session intent continuity without lexical repetition.
- Use the article only as supporting context.
- Favor novelty and depth over logistics unless the session
    explicitly shows planning intent.
```

This template specifies not only the prioritization of ORCID keywords but also the balance with session continuity and contextual information from the current article. In this way, the model is guided to integrate different signals in a controlled manner, while still ensuring novelty and avoiding simple repetition of past queries.

4.8 Model 3: Extended Synthetic Profiles

The third variant further expands personalization by enriching the user profiles with attributes that go beyond academic keywords. In addition to ORCID keywords, we created synthetic attributes for each user:

- **Nationality**, e.g., Italy, USA, India.
- **Profession**, e.g., Mathematics Professor, Physician, Student.
- **Personal interests and hobbies**, e.g., hiking, movies, museums, food.

These attributes were stored in CSV/Excel format, allowing us to systematically generate diverse user personas. The prompt was adapted accordingly: Gemini was instructed to also take into account the additional synthetic attributes (nationality, profession, and hobbies) alongside the ORCID keywords when generating the next query. The idea was to extend personalization beyond academic descriptors, ensuring that the model considered both research-related signals and broader aspects of a simulated user's identity.

Prompt template used in Model 3. For transparency and reproducibility, the exact instructions provided to Gemini were:

```
Guidance and priorities:
- Prioritize long-term user relevance from ORCID keywords and
    profession/persona.
- Maintain session intent continuity without lexical repetition.
- Use the article only as supporting context.
- Favor novelty and depth over logistics unless the session
    explicitly shows planning intent.
- ORCID Keywords have higher priority than Personal Entities;
  Use entities only as complementary, short-term signals.
```

In this way, the model is guided to produce suggestions that are both consistent with the user's research expertise and adapted to their broader profile.

4.9 Discussion

Together, these variants provide a clear picture of how personalization can evolve: from simply adding metadata, to carefully engineering prompts, to simulating realistic user profiles that reflect both academic background and personal identity.

This progressive design highlights an important insight: personalization is not only a matter of *what information* is available about the user, but also of *how this information is presented and weighted* in the interaction with the LLM. Similar observations have been made in the literature on prompt engineering and contextual recommendation [10, 9].

In the next chapter, these three models will be compared to assess how different levels of personalization and prompt design impact the quality of query suggestions.

Chapter 5

Evaluation

This chapter summarizes the research questions and outlines the evaluation procedure.

5.1 Research Questions

This study addresses the following research questions:

1. Does the inclusion of ORCID-based personalization (Model 1) improve query suggestions compared to the baseline?
2. How does the introduction of prompt weighting (Model 2) affect the quality of the generated suggestions?
3. How does the combination of enriched profiles and prompt weighting (Model 3) further shape personalization outcomes?

5.2 Evaluation Setup

As detailed in Chapter 4, user sessions were simulated using POI descriptions from Verona. User profiles combined both academic descriptors (ORCID keywords) and synthetic attributes such as nationality, profession, and hobbies.

The evaluation was conducted by **qualitatively comparing query suggestions** produced by the three models under identical input conditions. Instead of large-scale metrics, we focused on representative case studies. Each case study was designed to explore a different type of user profile, so that the models could be assessed across a variety of interests and contextual settings.

In addition, two of the scenarios intentionally lacked specific information:

- in one case, no ORCID keywords were available,
- in another, synthetic attributes were missing.

This allowed us to test whether the models were still able to produce coherent and relevant suggestions despite incomplete user profiles.

By combining diverse user interests with controlled cases of missing information, the evaluation provides insights into both the strengths and limitations of the proposed models.

5.3 Case Studies

Case Study 1: User with ORCID keywords and synthetic attributes

The user profile is defined as follows:

- **Profession:** Physician
- **Nationality:** Italy
- **Hobbies:** nature, hiking, pets, vegan food, wellness
- **ORCID keywords:** diabetes, metabolism, pancreatic beta cell function, insulin-resistance

The simulated interaction context consisted of:

- **Current query:** *"Best ice cream in Verona?"*
- **Session history:** *"Best ice cream in Verona?"*, *"Which is the cathedral of Verona?"*, *"Where can I buy opera tickets?"*, *"How can I visit Castelvecchio?"*, *"Best place for a traditional Veronese dinner?"*
- **Current article viewed:** *Gelateria La Romana*
- **Current article text:** Part of the La Romana artisanal chain, the Verona branch opened in 2013 on Piazza Santo Spirito 9. It pairs gelato made with a franchise-wide focus on ingredients with pastries and café service, offering both classic Italian flavors and rotating specials.
- **Extracted personal entities:** *verona, la romana, piazza santo spirito 9, italian, cafè.*

Given this context, the three models produced the following next-query suggestions:

Table 5.1: Example of query suggestions for a simulated physician profile.

Models	**Suggested Query**
1) ORCID keywords	"Gelateria Savoia Verona"
2) Prompt-weighted ORCID keywords	"Glycemic index of gelato ingredients"
3) Prompt-weighted enriched profiles	"Healthy traditional Veronese dishes"

Observations.

- **Model 1.** Suggestion: *"Gelateria Savoia Verona"*.
  Rationale: The user is clearly interested in ice cream in Verona, as shown by the initial query and the click on *Gelateria La Romana*. The article mentions that La Romana is part of a chain, implying that other options exist. Searching for another highly-rated gelateria in Verona, such as *Gelateria Savoia*, would allow the user to compare alternatives and potentially find another place to try. While the user's ORCID keywords indicate an interest in diabetes and metabolism, it's unlikely they are looking for sugar-free options given their current search behaviour. The session suggests a general interest in Verona and its attractions, further supporting the focus on finding good ice cream within the city.

- **Model 2.** Suggestion: *"Glycemic index of gelato ingredients"*.
  Rationale: This suggestion bridges the user's immediate interest in gelato (from the query, session, and current article) with their long-term research focus on diabetes and metabolism (ORCID keywords). It acknowledges the enjoyment of gelato while suggesting a deeper dive into its nutritional aspects, specifically the glycemic index, relevant to managing blood sugar levels. This adds a novel dimension to the search, moving beyond simply finding the "best" gelato to understanding its composition. It avoids direct repetition of previous searches but maintains thematic consistency with the current session's focus on food and Verona.

- **Model 3.** Suggestion: *"Healthy traditional Veronese dishes"*.
  Rationale: This query is suggested by combining your long-term professional interest in metabolism and diabetes (from your ORCID keywords) with your current session's focus on finding traditional food in Verona. After looking at a dessert like ice cream, this query helps you explore authentic local cuisine through a health-conscious lens, aligning with your personal interest in wellness without repeating previous searches.

This case illustrates how the three models progressively diverge in their behaviour. Model 1 remains anchored to the immediate context, suggesting another gelateria in Verona without considering the user's academic background. Model 2 successfully integrates long-term descriptors (ORCID keywords), shifting the recommendation from a purely touristic search to a medically relevant angle. Model 3, instead, broadens the perspective by leveraging profile enrichment and contextual continuity, moving beyond ice cream to suggest a different but related culinary experience. Together, these outputs highlight the incremental personalization achieved by combining prompt weighting and enriched profiles, and demonstrate how the system can balance session-based context with longer-term user characteristics.

It should be emphasized that the rationales presented above were automatically generated by Gemini and are included to illustrate the model's internal justification process, rather than being post-hoc interpretations by the author.

Case Study 2: User with ORCID keywords but no synthetic attributes

The user profile is defined as follows:

- **Profession:** — (no Profession declared)
- **Nationality:** — (no Nationality declared)
- **Hobbies:** — (no personal interests declared)
- **ORCID keywords:** Researcher, Sport Teacher, Handball and Bodybuilding coach, Sports Equipment Business

The simulated interaction context consisted of:

- **Current query:** *"UniVr"*

- **Session history:** *"UniVr"*, *"Best ice cream in Verona?"*, *"Most important church to visit in Verona?"*, *"Are there guided tours at the Roman Theatre museum?"*, *"Roman bridge with the best photo spot over the Adige"*, *"How to get from Verona Airport to the historic centre"*

- **Current article viewed:** *Università degli Studi di Verona*

- **Current article text:** Verona's public university offers undergraduate, graduate, and doctoral programs across multiple areas, with international tracks and seasonal schools.

- **Extracted personal entities:** *Verona.*

Given this context, the three models produced the following next-query suggestions:

Table 5.2: Example of query suggestions for a user without hobbies declared.

Models	**Suggested Query**
1) ORCID keywords	"University of Verona international programs"
2) Prompt-weighted ORCID keywords	"University of Verona sports facilities"
3) Prompt-weighted enriched profiles	"University of Verona sports facilities"

Observations.

- **Model 1.** Suggestion: *"University of Verona international programs"*.
Rationale: The user's initial query *UniVr* suggests they were looking for the University of Verona. The article confirms this. The session history indicates an interest in Verona, including tourist attractions. Combined with the article mentioning *international tracks and seasonal schools* and an *international programmes portal*, the next logical query might be to explore these programs specifically.

- **Model 2.** Suggestion: *"University of Verona sports facilities"*.
Rationale: The user's query *UniVr* and the article title *Università degli Studi di Verona* clearly indicate interest in the University of Verona. The session history shows an interest in Verona tourism, suggesting the user might be visiting or planning to visit. The ORCID keywords heavily emphasize sports-related topics *Sport Teacher,*

handball and Bodybuilding coach, Sports Equipment Business. Combining these, the next logical query might be about sports facilities at the university, catering to both the immediate interest in Verona and the long-term interest in sports. This suggestion prioritizes the user's ORCID keywords while maintaining session continuity by focusing on Verona. It also incorporates information from the article about the university and its distributed campuses.

- **Model 3.** Suggestion: *"University of Verona sports facilities"*.
  Rationale: The user's current query *UniVr* is short, likely indicating the start of their exploration, and the shown article confirms it refers to the University of Verona. The session history reveals an interest in Verona tourism, but the user's ORCID keywords strongly emphasize a background in sports and athletic training. Therefore, it's highly probable that the user's interest in the university relates to sports programs, facilities, or related research rather than general academic programs. This query combines the present context with the long-term user interest, prioritizing the ORCID keywords while respecting session continuity by remaining within the Verona context.

This case highlights that in the absence of additional synthetic attributes, Model 3 converges to the same behaviour as Model 2, as expected. Nevertheless, both models still ensure that ORCID-driven personalization is preserved, providing more relevant suggestions compared to Model 1, which ignores ORCID keywords.

Case Study 3: No ORCID keywords, only synthetic attributes

The user profile is defined as follows:

- **Profession:** Student
- **Nationality:** USA
- **Hobbies:** sports, NBA, movies, computer, beer
- **ORCID keywords:** — (Non-existent ORCID profile)

The simulated interaction context consisted of:

- **Current query:** *"What to do tonight"*
- **Session history:** *"What to do tonight"*, *"Best ice cream in Verona?"*, *"Where to drink something in the evening?"*, *"Which exhibition center hosts Vinitaly in Verona?"*, *"Best cinema in Verona"*

- **Current article viewed:** *Piazza Bra*
- **Current article text:** This square gives a magnificent glance at the history of the city: it is the foyer of the Arena on the occasion of the opera, which draws thousands of spectators.
- **Extracted personal entities:** *Verona, Arena, Piazza Bra.*

Given this context, the three models produced the following next-query suggestions:

Table 5.3: Example of query suggestions for a user without an ORCID profile.

Models	**Suggested Query**
1) ORCID keywords	"Events in Piazza Bra tonight"
2) Prompt-weighted ORCID keywords	"Events in Piazza Bra tonight"
3) Prompt-weighted enriched profiles	"Open air cinema Verona"

Observations.

- **Model 1.** Suggestion: *"Events in Piazza Bra tonight"*.
  Rationale: The user's initial query *What to do tonight* and their session history indicate an interest in evening activities in Verona. The article about Piazza Bra mentions that the square hosts many events. Combining the user's interest in evening activities, their focus on Verona and Piazza Bra, and the article's mention of events, *Events in Piazza Bra tonight* is a logical next query.
- **Model 2.** Suggestion: *"Events in Piazza Bra tonight"*.
  Rationale: The user's session clearly indicates an intent to find things to do in Verona, specifically focusing on evening activities *ice cream, drinks, cinema.* The current query *What to do tonight* reinforces this immediate need. The article about Piazza Bra describes it as a central location with events and activities. Combining the user's location interest (Verona, Piazza Bra from both session and article) with the time constraint ("tonight") and the article's mention of Piazza Bra hosting events, this query directly addresses the user's likely goal. While no ORCID keywords are provided, the suggestion strongly aligns with the session's established intent and leverages information from the visited article, making it the most probable next step.

- **Model 3.** Suggestion: *"Open air cinema Verona"*.
  Rationale: The user's session shows interest in evening entertainment in Verona *ice cream, drinks, cinema.* The article mentions Piazza Bra as a central location with events, including the Arena hosting opera. Combining these with the user's interest in *Movies*, an open-air cinema experience becomes a relevant suggestion, especially considering the likely pleasant summer evening weather implied by the initial query. Although the user has no ORCID keywords specified, movies is among the personal interests.

This case highlights that while Models 1 and 2 collapse to the same behaviour when ORCID data is absent, Model 3 still leverages synthetic attributes to produce a more tailored and contextually richer query suggestion.

5.4 Summary of Observations

The evaluation revealed several insights when considered in light of our Research Questions (RQ1–RQ3):

- **RQ1 – Does the inclusion of ORCID keywords improve personalization?** Model 1 showed limited personalization. In all cases, it produced valid but generic outputs, ignoring ORCID descriptors and relying mainly on session context and article entities. In contrast, Model 2 consistently integrated ORCID keywords into the suggestions. When such descriptors were available, the model generated more domain-specific and relevant queries (e.g., medical aspects of food, sports facilities), clearly improving personalization compared to Model 1.

- **RQ2 – How does prompt design affect the use of personal descriptors?** Model 2 demonstrated that explicitly weighting ORCID descriptors in the prompt ensures they are systematically incorporated into the generated queries. This guidance prevented the model from overlooking long-term user information, which was a limitation observed in Model 1. Model 3 further confirmed the central role of prompt engineering: by adapting the instructions to account not only for ORCID descriptors but also for synthetic profile attributes, it consistently balanced short-term session context with longer-term user signals.

- **RQ3 – Can enriched synthetic profiles further extend personalization?** Model 3 exhibited the strongest personalization capacity. It consistently produced the most diverse and tailored outputs, blending professional, academic, and personal aspects. Even in scenarios with incomplete information (e.g., missing ORCID keywords), it leveraged synthetic attributes to generate contextually appropriate and personalized suggestions.

It should also be emphasized that the degree of personalization is conditioned by the availability of user signals. In practice, ORCID keywords may be absent if researchers do not actively curate their profiles, if certain disciplines make limited use of ORCID, or if individuals deliberately withhold information for privacy reasons.
Similarly, synthetic attributes such as hobbies or nationality may be missing due to incomplete datasets or the user's decision not to disclose personal details. These situations explain why some profiles rely primarily on short-term session entities, whereas others can benefit from richer long-term descriptors. Understanding these constraints is essential, as it highlights both the potential and the limitations of the proposed models, and provides the foundation for the overall discussion presented in Chapter **??**.

Chapter 6

Conclusions

This thesis addressed the problem of **personalized contextual query suggestion**, focusing on the integration of structured user information into a lightweight framework inspired by K-LaMP [8]. The central contribution lies in the **design, implementation, and comparison** of three models that progressively enrich the personalization process:

- **Model 1:** baseline personalization using ORCID keywords appended to the context.
- **Model 2:** prompt-weighted personalization, explicitly prioritizing ORCID descriptors.
- **Model 3:** enriched personalization combining ORCID keywords with synthetic profile attributes such as nationality, profession, and hobbies.

6.1 Summary of Findings

This thesis has demonstrated that **entity-centric personalization** can enhance query suggestion by integrating ORCID profiles and contextual information. The evaluation, conducted on simulated sessions with Points of Interest (POIs) in Verona, shows that the three proposed models progressively increase the degree of personalization.
Model 1 produced valid but often generic suggestions, revealing the limitations of simply appending ORCID keywords to the context.
Model 2 demonstrated that prompt weighting enables ORCID descriptors to guide the system towards more domain-relevant and academically aligned queries.
Model 3, by incorporating synthetic profile attributes, generated more diverse and user-centred results, particularly when lifestyle or professional

information was available.

Overall, the evaluation demonstrated that entity-centric knowledge and structured user descriptors lead to more tailored and contextually relevant suggestions. In addition, the system proved robust in scenarios where user information was incomplete. Even when ORCID keywords or synthetic attributes were missing, the models were still able to fall back on short-term session signals and produce coherent outputs, although richer personalization was achieved whenever long-term descriptors were available.

6.2 Strengths and Limitations

Strengths.

- **Reproducibility:** the framework is implementable without access to proprietary search logs, relying instead on publicly available POI data and synthetically generated profiles.
- **Flexibility:** the modular design of the pipeline (memory stream, entity store, prompt assembly) allows straightforward adaptation to different domains and user descriptors.
- **Transparency:** entity extraction and prompt construction explicitly indicate which user signals are leveraged at each stage, thereby enhancing interpretability.
- **Scalability of personalization:** the three models illustrate how personalization can be incrementally enriched, from simple academic descriptors to broader user personas.

Limitations.

- **Synthetic evaluation setting:** the study relies on simulated sessions with POIs in Verona, rather than on real-world search logs or user interactions. This restricts the realism of the evaluation and limits the ability to capture genuine user behaviour.
- **Qualitative assessment:** personalization is assessed through a small number of case studies and qualitative inspection, rather than through automatic metrics or large-scale user trials. As a result, findings should be interpreted as illustrative rather than statistically conclusive.

- **Domain bias:** Because the evaluation is based on Verona POI descriptions, the results are oriented toward cultural and touristic queries. This introduces a contextual bias and may limit generalizability to other domains such as academic search, e-commerce, or healthcare.
- **Dependence on user descriptors:** when ORCID keywords or synthetic attributes are missing, personalization is constrained, forcing the system to fall back on short-term contextual signals (e.g., current query, session history, article content). This highlights the dependency on the richness of user profiles for effective personalization.
- **Potential risk of over-personalization:** Over-personalization may risk misalignment with a user's immediate intent. In practice, however, the models often prioritized coherence over strict adherence to long-term descriptors, thus mitigating this risk. Future research may wish to study this balance more explicitly.

6.3 Future Directions

Several directions emerge for future work:

- **Automatic evaluation metrics:** develop scalable proxies for measuring relatedness, novelty, and usefulness of query suggestions, with the goal of complementing or replacing manual inspection.
- **Cross-domain generalization:** extend experiments beyond cultural tourism in Verona, testing the robustness of the framework in other domains such as academic search, e-commerce, or healthcare.
- **Real-world deployment:** integrate the pipeline into interactive systems (e.g., recommender platforms or conversational agents) to validate its effect on real user engagement and satisfaction.
- **Privacy-aware design:** explore and implement privacy-preserving approaches (e.g., anonymization) to ensure the responsible use of user descriptors.

In summary, future work should aim to broaden the scope of evaluation, assess the framework across diverse application domains, and develop privacy-preserving implementations. Pursuing these directions would consolidate the contribution of this thesis and bring knowledge-augmented personalization closer to real-world deployment.

6.4 Concluding Remarks

Main Findings

This thesis has presented a reimplementation and extension of the K-LaMP framework for personalized contextual query suggestion. By integrating entity-centric knowledge stores with both academic descriptors (ORCID keywords) and synthetic user attributes (such as profession, nationality, and hobbies), the study has demonstrated how personalization can be progressively enriched and controlled through prompt design.

The experimental evaluation highlighted three main findings:

- The mere inclusion of ORCID keywords, without explicit prioritization, does not substantially differentiate the output from a baseline model.
- Prompt engineering is crucial in steering large language models towards domain-relevant and user-centred suggestions.
- Extending user profiles with synthetic attributes enables the system to capture broader dimensions of user identity, producing more diverse and realistic recommendations.

Dependence on User Feedback

These results underline that **the effectiveness of personalization depends on the availability and prioritization of user signals**. When ORCID data or synthetic attributes are missing, the framework still provides coherent next-query recommendations, but it relies more heavily on contextual signals, such as the current query, session history, and article content. In contrast, when richer user profiles are available, the system can leverage them to deliver suggestions that more closely reflect long-term expertise, personal interests, and lifestyle dimensions.

Contributions of this Thesis

The contributions of this thesis are twofold:

- **Personalization beyond proprietary logs:** The original K-LaMP framework relied on large-scale Bing logs, which are not publicly available and raise privacy concerns. In contrast, this work shows that meaningful personalization can be achieved using openly accessible or synthetically generated data (POI descriptions, ORCID keywords, and enriched user profiles). This provides a reproducible and privacy-preserving pathway for research, accessible even without industrial-scale datasets.

- **Prompt design as a central driver:** Simply appending user keywords to the context produces only limited effects. By contrast, explicitly guiding the model to prioritize these signals (Model 2) or combining them with broader attributes such as nationality, profession, or hobbies (Model 3) leads to markedly more personalized suggestions. This indicates that personalization depends not only on the availability of user signals but also on how these are framed, weighted, and communicated through the prompt.

Implications

Together, these contributions represent a concrete step towards **adaptive, interpretable, and user-centred query recommendation systems**: adaptive, as the pipeline can operate across heterogeneous profiles with different levels of available data; interpretable, since the entity knowledge store and prompt design clearly reveal which signals are being used; and user-centred, by integrating not only academic descriptors but also lifestyle attributes, reflecting the multifaceted identity of real users.
In conclusion, this thesis **paves the way** for future research and applications in knowledge-augmented personalization, demonstrating how entity-centric signals and prompt design can be combined to move beyond generic query suggestions and towards richer, context-aware personalization.

Acknowledgements

At the conclusion of this thesis, I would like to acknowledge all the people without whom this work would not have taken the same form.

I sincerely thank my supervisor, Prof. Elisa Quintarelli, who during these five months of work, has guided me with constant support and practical suggestions in both research and writing. I also thank my co-supervisor, Prof. Niccolò Marastoni, for his valuable feedback and support that helped refine this work.

I am deeply grateful to my parents for always standing by me, even in moments of doubt or discouragement, and for enabling me to complete my studies by supporting me both practically and emotionally.

I would also like to thank my friends, who have supported me throughout these years of study and shared with me both the challenges and the satisfactions of this academic journey.

A special thanks goes to my friend Enrico, for his invaluable support and encouragement during the final months of writing this thesis.

Finally, I would like to dedicate this small achievement to myself, for the perseverance and commitment I have always shown in bringing to completion what I started, despite the challenges encountered along the way, in the hope that it may serve as encouragement for pursuing future goals.

Bibliography

[1] B. J. Jansen, A. Spink, J. Bateman, and T. Saracevic, "Real life information retrieval: A study of user queries on the web," *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, vol. 32, no. 1, pp. 5–17, 1998.

[2] A. Feuer, S. Savev, and J. A. Aslam, "Evaluation of phrasal query suggestions," in *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, ser. CIKM '07. New York, NY, USA: Association for Computing Machinery, Nov. 2007, pp. 841–848.

[3] D. Kelly, K. Gyllstrom, and E. W. Bailey, "A comparison of query and term suggestion features for interactive searching," in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '09. New York, NY, USA: Association for Computing Machinery, Jul. 2009, pp. 371–378.

[4] R. Jones, B. Rey, O. Madani, and W. Greiner, "Generating query substitutions," in *Proceedings of the 15th International Conference on World Wide Web*, ser. WWW '06. New York, NY, USA: Association for Computing Machinery, May 2006, pp. 387–396.

[5] S. Bhatia, D. Majumdar, and P. Mitra, "Query suggestions in the absence of query logs," in *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. Beijing China: ACM, Jul. 2011, pp. 795–804.

[6] OpenAI, "Gpt-4 technical report," 2023. [Online]. Available: https://arxiv.org/abs/2303.08774

[7] Google DeepMind, "Gemini: A family of highly capable multimodal models," *arXiv preprint*, 2023. [Online]. Available: https://arxiv.org/abs/2312.11805

[8] J. Baek, N. Chandrasekaran, S. Cucerzan, A. herring, and S. K. Jauhar, “Knowledge-augmented large language models for personalized contextual query suggestion,” 2024. [Online]. Available: https://arxiv.org/abs/2311.06318

[9] S. Geng, S. Liu, Z. Fu, Y. Ge, and Y. Zhang, “Recommendation as language processing (p5): A unified pretrain, personalized prompt & predict paradigm,” *ACM Conference on Recommender Systems (RecSys)*, 2022.

[10] J. Ji, Z. Li, S. Xu, W. Hua, Y. Ge, J. Tan, and Y. Zhang, “Genrec: Large language model for generative recommendation,” *arXiv preprint*, 2023. [Online]. Available: https://arxiv.org/abs/2307.00457

[11] W.-C. Kang, J. Ni, N. Mehta, M. Sathiamoorthy, L. Hong, E. Chi, and D. Z. Cheng, “Do LLMs Understand User Preferences? Evaluating LLMs On User Rating Prediction, shorttitle = Do LLMs Understand User Preferences?” May 2023.

[12] H. Lyu, S. Jiang, H. Zeng, Y. Xia, and J. Luo, “Llm-rec: Personalized recommendation via prompting large language models,” *arXiv preprint*, 2024. [Online]. Available: https://arxiv.org/abs/2307.15780

[13] M. Komeili, K. Shuster, and J. Weston, “Internet-augmented dialogue generation,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021, pp. 8460–8478.

[14] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, “Personalizing dialogue agents: I have a dog, do you have pets too?” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018, pp. 2204–2213.

[15] C. Li, M. Zhang, Q. Mei, Y. Wang, S. A. Hombaiah, Y. Liang, and M. Bendersky, “Teach LLMs to Personalize – An Approach inspired by Writing Education,” Aug. 2023.

[16] F. Yang, Z. Chen, Z. Jiang, E. Cho, X. Huang, and Y. Lu, “PALR: Personalization Aware LLMs for Recommendation,” Jun. 2023.

[17] P. Resnick and H. R. Varian, “Recommender systems,” in *Communications of the ACM*, vol. 40, no. 3. ACM, 1997, pp. 56–58.

[18] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds., *Recommender Systems Handbook*. Springer, 2011.

[19] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[20] R. Burke, "Hybrid recommender systems: Survey and experiments," in *User Modeling and User-Adapted Interaction*, vol. 12, no. 4. Springer, 2002, pp. 331–370.

[21] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 1877–1901. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

[22] L. Reynolds and K. McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm," 2021. [Online]. Available: https://arxiv.org/abs/2102.07350

[23] A. Sordoni, Y. Bengio, H. Vahabi, C. Lioma, J. G. Simonsen, and J.-Y. Nie, "A hierarchical recurrent encoder-decoder for generative context-aware query suggestion," in *Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM)*, 2015, pp. 553–562.

[24] M. Dehghani, S. Rothe, E. Alfonseca, and P. Fleury, "Learning to attend, copy, and generate for session-based query suggestion," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM)*, 2017, pp. 1747–1756.

[25] Q. Cheng, Z. Ren, Y. Lin, P. Ren, Z. Chen, X. Liu, and M. de Rijke, "Long short-term session search: Joint personalized reranking and next query prediction," in *Proceedings of the Web Conference (WWW)*, 2021, pp. 239–248.

[26] M. Lissandrini, A. P. d. Vries, and J. Hidders, "Graph-query suggestions for knowledge graph exploration," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. ACM, 2020, pp. 1281–1296.

[27] N. Craswell and M. Szummer, "Random walks on the click graph," in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. ACM, 2005, pp. 239–246.

[28] M. Cataldi, L. Di Caro, and C. Schifanella, "Personalized query expansion using community-based relevance feedback," in *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2010, pp. 345–352.

[29] K. Lee, H. Yang, and S.-W. Bae, "Boomerang: Proactive insight-based recommendations for data exploration," in *Proceedings of the 2021 ACM SIGMOD International Conference on Management of Data*. ACM, 2021, pp. 2502–2514.

[30] G. Marchionini, "Exploratory search: From finding to understanding," *Communications of the ACM*, vol. 49, no. 4, pp. 41–46, 2006.

[31] N. J. Belkin, P. G. Marchetti, and C. Cool, "Situated information seeking and interactive information retrieval," in *Proceedings of the 1992 ACM SIGIR Conference on Research and Development in Information Retrieval*, 1992, pp. 11–22.

[32] A. Bacciu, E. Palumbo, A. Damianou, N. Tonellotto, and F. Silvestri, "Generating query recommendations via llms," 2024. [Online]. Available: https://arxiv.org/abs/2405.19749

[33] K. Morton, M. Balazinska, D. Grossman, and J. Mackinlay, "Support the data enthusiast: Challenges for next-generation data-analysis systems," *Proceedings of the VLDB Endowment*, vol. 7, no. 6, pp. 453–464, 2014.

[34] J. Heer and B. Shneiderman, "Interactive dynamics for visual analysis," *Communications of the ACM*, vol. 55, no. 4, pp. 45–54, 2012.

[35] W. U. Ahmad, K.-W. Chang, and H. Wang, "Context attentive document ranking and query suggestion," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2019, pp. 385–394.

[36] T. Luera, X. Chen, R. Gupta, and Y. Zhang, "Optimizing data delivery: Insights from user preferences on visuals, tables, and text," in *Proceedings of the ACM Web Conference (WWW)*. ACM, 2025, pp. 1124–1136.

[37] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 423–443, 2018.

[38] D. Ceneda, T. Gschwandtner, T. May, S. Miksch, and H.-J. Schulz, "Characterizing guidance in visual analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 111–120, 2017.

[39] D. Barman, R. Sarkar, and N. Chowdhury, "A cooperative co-evolutionary genetic algorithm for query recommendation," *Multimedia Tools and Applications*, vol. 83, no. 9, pp. 11 461–11 491, 2024.

[40] B. M. Fonseca, P. B. Golgher, E. S. de Moura, and N. Ziviani, "Using association rules to discover search engines related queries," in *Proceedings of the 1st Latin American Web Congress (LA-WEB)*. IEEE, 2003, pp. 66–71.

[41] T.-Y. Liu, "Learning to rank for information retrieval," *Foundations and Trends in Information Retrieval*, vol. 3, no. 3, pp. 225–331, 2009.