Representing, tracking, and evaluating user's changing knowledge and needs in information retrieval

Dima El Zein

UNIVERSITÉ CÔTE D'AZUR

ÉCOLE DOCTORALE
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THÈSE DE DOCTORAT

Représenter, Suivre et Evaluer les Connaissances et Besoins des Utilisateurs et leur Evolution dans le Cadre de la Recherche d'Information

Dima EL ZEIN

Laboratoire d'Informatique, de Signaux et Systèmes de Sophia Antipolis (I3S)
UMR7271 UCA CNRS

Présentée en vue de l'obtention du grade de docteur en Informatique **d'**Université Côte d'Azur

Dirigée par : Célia DA COSTA PEREIRA, Maîtresse de Conférences, HDR, Université Côte d'Azur

Soutenue le : 10 Juillet 2023

Devant le jury, composé de :
Elena CABRIO, Professeure, Université Côte d'Azur
Lynda TAMINE-LECHANI, Professeure, Université Toulouse III - Paul Sabatier
Justin ZOBEL, Professeur, University of Melbourne
Kevyn COLLINS-THOMPSON, Professeur, University of Michigan

i3s sophia antipolis

REPRÉSENTER, SUIVRE ET EVALUER LES CONNAISSANCES ET BESOINS DES UTILISATEURS ET LEUR EVOLUTION DANS LE CADRE DE LA RECHERCHE D'INFORMATION

Representing, Tracking, and Evaluating User's Changing Knowledge and Needs in Information Retrieval

Dima EL ZEIN

Jury :

Président du jury

Elena CABRIO, Professeure, Université Côte d'Azur

Rapporteurs

Lynda TAMINE-LECHANI, Professeure, Université Toulouse III - Paul Sabatier
Justin ZOBEL, Professeur, University of Melbourne

Examinateurs

Kevyn COLLINS-THOMPSON, Professeur, University of Michigan

Directrice de thèse

Célia DA COSTA PEREIRA, Maîtresse de Conférences, HDR, Université Côte d'Azur

---

Université Côte d'Azur

Dima EL ZEIN

Représenter, Suivre et Evaluer les Connaissances et Besoins des Utilisateurs et leur Evolution dans le Cadre de la Recherche d'Information

xii+163 p.

Ce document a été préparé avec LaTeX2e et la classe `these-ISSS` version v. 2.10.

Impression : output.tex – 13/11/2023 – 21:42

Révision pour la classe : these-ISSS.cls,v 2.10 2020/06/24 14:16:37 mpelleau

Représenter, Suivre et Evaluer les Connaissances et Besoins des Utilisateurs et leur Evolution dans le Cadre de la Recherche d'Information

Résumé

L'utilisation de systèmes de recherche d'information est désormais un élément essentiel de notre quotidien, offrant une source d'information riche et facile d'accès. Ces systèmes, notamment les moteurs de recherche, peuvent maintenant fournir rapidement des données factuelles en adaptant les résultats en fonction de certains facteurs contextuels tels que la localisation, le type d'appareil et les intérêts de l'utilisateur. Cependant, ils ne sont optimisés, ni pour répondre aux objectifs d'apprentissage des utilisateurs, ni pour tenir compte de l'évolution de leurs connaissances. En effet, les connaissances de l'utilisateur ne sont pas statiques, mais bien dynamiques : l'utilisateur entame une session de recherche avec ses connaissances préexistantes et il continue d'en acquérir de nouvelles tout au long du processus de recherche. Lors de l'utilisation des outils de recherche actuels, l'utilisateur peut soumettre plusieurs requêtes et ainsi examiner de nombreux documents, dont certains peuvent manquer de pertinence et n'apporter aucune plus-value à sa connaissance. Cela peut entraîner une perte de temps et de motivation dans le processus de recherche d'informations souhaitées. C'est pour cela que le domaine de l'adaptation des systèmes de recherche pour l'apprentissage de connaissances, communément appelé "recherche comme apprentissage" ou "search as learning", a récemment fait l'objet d'une attention considérable. Cependant, nous pensons qu'avant de nous lancer dans l'adaptation de tels systèmes, il est tout d'abord indispensable de comprendre comment les utilisateurs apprennent et acquièrent des connaissances et de savoir comment les informations doivent être structurées dans les systèmes. Par ailleurs, l'évaluation de ces systèmes présente un réel défi car les méthodes de recherche et les mesures d'évaluation existantes négligent souvent la représentation et le suivi des connaissances de l'utilisateur. De plus, aucun jeu de données n'est disponible pour mesurer l'efficacité de ces systèmes qui prétendent aider à l'apprentissage pendant les sessions de recherche. Dans cette thèse, notre objectif est de surmonter ces problèmes en proposant différentes approches pour représenter les connaissances de l'utilisateur ainsi que ses objectifs d'apprentissage dans les systèmes de recherche. Nous proposons un cadre capable de suivre de manière dynamique l'évolution de ses connaissances et de ses besoins et d'estimer l'évolution de l'apprentissage tout au long de la session de recherche. Nous proposons ensuite une nouvelle mesure qui évalue dynamiquement les documents et les classent selon les besoins changeants de l'utilisateur et l'évolution de ses connaissances. Nous construisons également un jeu de données permettant de suivre l'évolution des connaissances de l'utilisateur tout au long des sessions de recherche. Ce jeu de données pourra servir de référence pour de futurs travaux. Enfin, nous proposons un cadre théorique pour implémenter ces concepts dans un système multi-agents doté de capacités de raisonnement basées sur des règles.

Mots-clés : Recherche d'information, Recherche comme Apprentissage, État de Connaissance, Objectifs d'Apprentissage, Évaluation, Agents BDI.

Representing, Tracking, and Evaluating User's Changing Knowledge and Needs in Information Retrieval

Abstract

The use of search tools has become an integral part of our daily lives, providing a readily accessible source of information and facilitating our acquisition of knowledge. However, while these tools are efficient in delivering factual data and adapting results based on contextual factors such as location, device, and interests, they are not optimized to support users' learning goals or account for their changing knowledge states. The user's knowledge is not static but rather dynamic, as they enter a search session with existing knowledge and continue to acquire knowledge during the search process. When using current search tools for learning purposes, a user might have to submit multiple queries and review numerous documents, some of which may lack relevance, resulting in increased time consumption in finding the desired information. The field of adapting search systems for learning objectives, commonly known as "search as learning", has recently gained significant attention. However, we argue that before adapting these systems to support learning, it is crucial to understand how users learn and acquire knowledge, and how to structure this information in the systems. Additionally, evaluating these systems presents challenges as existing retrieval methods and evaluation measures often overlook the representation and tracking of users' knowledge. Furthermore, there is a lack of available datasets for measuring the effectiveness of systems that claim to support learning during search sessions. In this thesis, we aim to overcome these issues by, firstly, exploring and proposing various approaches for representing users' learning goals in search systems. We propose a framework that can dynamically track the users' evolving knowledge and needs and estimate the user's learning outcome at any time during the search session. Secondly, we propose a novel evaluation measure that dynamically evaluates documents and ranked lists of documents with respect to changing user needs and evolving knowledge. Furthermore, we construct a dataset to track the evolution of the users' knowledge throughout the search sessions, which will serve as a benchmark for other researchers. Finally, we propose a theoretical framework for implementing these concepts in an agent-based system with rule-based reasoning capabilities.

Keywords: Information Retrieval, Search as Learning, Knowledge State, Learning Objectives, Evaluation, BDI Agents.

Acknowledgment

As I approach the end of my PhD journey, I reflect on the beginning : a brief 12-hour trip to France for selection interview and, after being accepted, leaving my home country, family, and job. It was a remarkable step taken towards this goal and also a new beginning in my life. This journey was filled with diverse and enriching experiences including numerous publications, presentations, conferences, travels, scientific collaborations, grants, scientific mediation, competitions, awards ... and adapting to the challenges of a pandemic. Each of these experiences - pandemic lockdowns aside - was enjoyed thoroughly and would have been impossible without the support of my supervisor, Célia da Costa Pereira.

Right from the beginning of this journey, Célia's guidance has been there, and it remains as I prepare to embark on the next one. She has been more than a supervisor, she has been a mentor, standing beside me during moments of doubt and always believing in my potential. Working under her guidance, whether it was conducting research, co-authoring publications, or teaching, has been a thoroughly enjoyable experience. Célia, you are the ideal supervisor that every PhD student hopes to have.

I am equally grateful to Andrea Tettamanzi, whose office door was always open for insightful discussions and brainstorming sessions. His external perspective was invaluable from the early phases of my journey through to the defense preparation. Among the several things he helped me with, I'll never forget when he and Célia helped me with carrying that big monitor to my place just one day before the lockdown. Thank you, Andrea, for your support.

I must also thank Alain Giboin and Anne-Marie Pinna for offering advice on our experimental designs. Their opinions, seeing things from different angles, taught me a lot, particularly in experimental psychology, ergonomics, and human-computer interaction.

My participation in conferences has profoundly enriched my academic journey and my outlook on my research. I am grateful to the welcoming community of Information Retrieval and Search as Learning for their openness to discussions and their readiness to provide advice. I am thankful for the connections I've made with individuals from across the globe at these conferences. The countless enjoyable dinners, lively discussions, and the friendships that blossomed from these gatherings have been a highlight of my participation. While it is impossible to name everyone, I must mention Nilavra Bhattacharya and Yassine Ghafourian, who are working on very similar topics of mine ; our conversations, both online and in person, were always a delight. I met Arthur Câmara from the University of Delft at one of these events, which led to a collaboration resulting in two publications ; our teamwork was a delightful experience. Thanks also to Magali Richir and Sabine Barrere for their exceptional assistance in organizing all the travel logistics.

It was at one of those early conferences where I met Professor Justin Zobel. We had an extensive discussion following his talk, whom I found incredibly humble, particularly when he expres-

sed interest in my topic and offered encouragement. I am honored that, years later, he became the reviewer of my thesis.

I had also the privilege of meeting Professor Kevyn Collins-Thompson, a connection that began with a simple email request for access to a dataset from his research and flourished into an academic collaboration. Collaborating on a project of common interest was a rewarding experience, and it taught me many new things. I am deeply thankful for the opportunity he provided to visit his group at the University of Michigan for a few months. His welcome, as well as that of his group, was incredibly warm and encouraging. These final months working on this project brought a sense of completion to my PhD journey.

Professor Lynda Tamine-Lechani, whose work on contextual search provided a solid foundation for my initial readings and exploration of my topic, has been a significant inspiration for my work. I am honored to have her on my jury and to receive her feedback. Additionally, I am grateful to Professor Elena Cabrio for serving as the president of my jury and for her organization and support.

Teaching has been both new and rewarding for me. My classes were as much a learning opportunity for me as they were for my students. I hope I have inspired a similar passion for programming and computer science in you, and I am grateful for your trust. I'm also thankful for Cathy Escazut for flexibly assigning many hours of her courses, which made it possible for me to fulfill my teaching hours with ease.

The friendships I made in the lab are cherished, as are the relieving coffee breaks. Special thanks to Thibaut and Tim for their presence and support when needed, the many meals we shared, to Amine my first lab friend, for the enriching and existential discussions, and to Maroua for the post-work workouts that provided a refreshing escape and good company.

I feel truly blessed with friends from various cities and countries. Fatima, whose presence I felt despite the distance; Yara, Alice, and May, who were always there for a comforting call, needed or not; Rima and Rashad, immensely helpful when I first arrived in France; Soha, my long-time study buddy, for her help and proofreading my dissertation; and my cousins, Sabine and Lynn, for their daily support. Sacha, whom I met upon arriving in France, quickly became like a little sister to me, staying by my side through tough times. Additionally, my extended family in Lebanon, including more cousins, uncles, and aunts, ensured that my parents were surrounded by love and support. And finally, to Souha, who provided immense support and was consistently helpful at every step and in every detail, all while managing her own demanding research path.

In closing, I turn to my parents, whose professional success and modesty have been a constant source of inspiration. My father, whose absence on the day of my defense is deeply felt, instilled in me a sense of self-belief that continues to guide me. My mother, ever the advocate for excellence, always pushed me to reach beyond the perfect score, teaching me that success knows no bounds. Finally, to my sister, Noura, who provided all the necessary support, including academically, to the extent that she could have recited my work at the defense and almost answered questions.

Finally, I thank each and every individual who has played a role, big or small, in my academic journey and personal development. Your support and encouragement have been the foundation upon which I have built my achievements

Table of Contents

Contributions

CHAPTER 1

Introduction

1.1 Motivation and context

Information Retrieval (IR) systems, including search tools, help users find information within vast amounts of data, in particular on the Web. Users resort to an IR system driven by their motivation to satisfy a specific informational need or goal. Belkin has defined this particular state in the user's cognition as an Anomalous State of Knowledge (ASK) (Belkin, Oddy, & Brooks, 1982) that is characterized by uncertainty or confusion arising from a knowledge gap, internal inconsistency, or conflicting evidence. Users typically express their need by submitting a query, which is a set of terms, to the search system (Borlund, 2003a). The search system responds to the query by typically returning a list of ranked documents that it deems relevant to the user's query. Traditionally, the system would return the same list of documents or pages for the same query, regardless of the user who submitted the query.

Search activities are not limited to simple fact-finding or navigational tasks (Marchionini, 2006); they can also include open-ended and informational tasks (Russell, Tang, Kellar, & Jeffries, 2009). Informational search sessions involve an intention to learn, as users desire to acquire knowledge or information about a particular topic that is typically spread across multiple pages. In recent years, it has become increasingly common for users to use retrieval systems, such as Web search engines, as learning tools. Users rely on these systems in their everyday lives to access information and learn (Selwyn, 2008; Biddix, Chung, & Park, 2011). While current search engines, specifically commercial ones, perform well at retrieving information quickly and responding effectively to factual or simple tasks, they are not optimized for learning and complex tasks (Hassan Awadallah, White, Pantel, Dumais, & Wang, 2014). These engines are typically optimized to increase sales of paid advertisements, query-document relevance, or popularity (Machado, de Alcantara Gimenez, & Siqueira, 2020). This does not mean that users do not learn when using search engines, but they often have to reformulate several queries and read many documents, some of which might be irrelevant or redundant to what they already know or what has been previously proposed. Not being able to find the necessary information can be time-consuming, as users may end up spending more time trying to locate the information they need (Brand-Gruwel, Wopereis, & Walraven, 2009). In such cases, they may opt to abandon the search engine and switch to more structured sources of information, such as experts, forums, online courses, or other resources.

The field of information retrieval has recently been actively researching the intersection between information retrieval, education, and psychology under an emerging field known as *search as learning* (SAL) (Collins-Thompson, Rieh, Haynes, & Syed, 2016). This field aims at investigating and adapting search systems for learning objectives. Recent work in SAL has encompassed a wide scope of efforts, ranging from adapting retrieval algorithms (Collins-Thompson, Hansen, & Hauff, 2017) and frameworks (Collins-Thompson & Callan, 2004), to designing search tools

and scaffolding for users (Câmara, Roy, Maxwell, & Hauff, 2021a; Câmara, Maxwell, & Hauff, 2022a). An area of research that has been particularly active is the prediction of the user's knowledge gain or learning outcome by the end of a search session using some behavioral features of the search session (Gadiraju, Yu, Dietze, & Holtz, 2018; R. Yu, Gadiraju, Holtz, et al., 2018a; J. Liu, Liu, & Belkin, 2016a). One of the long-term goals of the SAL field is personalizing the search results based on the user's knowledge state and learning objectives.

Personalization of communication is intuitive in human interactions. When we talk to others, we often adapt our language and the level of detail we provide to match the other person's knowledge and understanding. For example, consider a father who is asked by his 16-year-old and 8-year-old children about *how car engines work*. The father may provide two different answers tailored to each child's age and level of knowledge. To the 16-year-old, the father might explain that *"A car engine is a complex system that uses internal combustion to generate energy. When you turn the key, the starter motor activates, initiating combustion. Fuel and air mix and ignite, creating an explosion that moves the pistons up and down. The crankshaft converts this motion into rotary motion, powering the car's wheels."*. On the other hand, the father may explain to the 8-year-old that the engine in a car is like a *"A car engine is like the heart of a car. Inside the engine, there is a mixture of fuel and air. When they come together, they create an explosion. This explosion makes certain parts of the engine move, kind of like how your heart pumps blood. These moving parts help the car's wheels turn, allowing it to move and take us where we want to go."*. Although both answers are relevant to the asked question, one answer might not be convenient if it were provided for the other child.

When users pose the same question to search engines, the system is likely to return the same answer to all users, regardless of their age or level of knowledge. Search engines operate based on generic relevance rather than accounting for the user's specific context, knowledge, and interests. As a consequence, users often obtain results that are not precisely what they were looking for, which can lead to frustration and a waste of time.

We aim to provide search systems the ability to personalize results based on the user's knowledge, which requires finding methods to represent the user's knowledge in an understandable format. Representing what the user knows and who they are is an important step in adapting search systems to help users learn, and store this information in a machine-understandable structure. This is known as user profiling. While user profiling is not a new field in information retrieval, as current systems personalize their search results based on factors such as geographical location, language, or interests, little research has been done on profiling users based on their knowledge and learning needs.

In this thesis, we differentiate between the user's knowledge and their interests because they are two different concepts. We draw inspiration from how users are profiled for their interests to explore efficient ways of representing the user's knowledge. The main difference between these two concepts is that knowledge is more dynamic than interest, as it changes every time the user is exposed to new information or reads a document. Representing the knowledge also requires a more granular representation to accurately assess what the user knows and what they do not. By accurately tracking the user's knowledge state, a search system can estimate a quantification of the learning outcome in relation to the learning objective, and recommend appropriate content. While user profiling is an essential step in understanding the user's knowledge state, it is not sufficient, as this knowledge changes rapidly when the user is exposed to new information. This means that the user's profile must be continually tracked and updated as it evolves.

We propose a framework for search as learning that can represent the user's knowledge, track its changes, and estimate the learning outcome at any given point in time. We explore various representation models for the user's knowledge and learning objectives and demonstrate the accuracy of the framework in estimating the user's real knowledge gain. This may be determined by calculating the difference between the user's knowledge level after the session and their knowledge level before the session.

Evaluating the performance of search systems that take into account the search context has been a challenging task. This is because there has been no standard evaluation established yet or traditional evaluation measures are still being used. In the search as learning domain, where the user's knowledge serves as a contextual factor, this challenge particularly persists especially due to the dynamic nature of users' knowledge (Brookes, 1980). Traditionally, the relevance of a document was measured based on its matching with the topic of the query, i.e., topical relevance, and the same list of results was returned to all users. However, the introduction of new factors in the IR process through search as learning indicates that assessing the relevance of search results based solely on query-document match is no longer sufficient. To overcome this challenge, we introduce a new definition of relevance and propose a personalized evaluation measure accordingly. The measure considers each user's knowledge state and their progress toward their learning objectives when assessing the relevance of search results.

Another particularity that makes evaluating search as learning systems challenging is the lack of appropriate datasets that can serve as a benchmark for a baseline. In order to address this challenge, we propose an adapted dataset that can serve as a valuable resource for evaluating future retrieval algorithms of personalized learning. This dataset consists of a set of queries and a set of relevant documents for each query, along with an estimated relevance for learning score for every document. Additionally, it includes an estimated relevance score for learning associated with every document. The dataset is publicly available.

One characteristic of humans is their ability to reason, which means their knowledge extends beyond what they may read in documents or other sources. Ideally, a system should be capable of inferring this knowledge coming from reasoning as well. When a system is aware of a user's knowledge, it should have the ability to deduce additional knowledge based on what it already knows about the user. To incorporate this reasoning process into the system, we leverage the advantages of an agent structure, as rational agents possess reasoning capabilities. We should also be able to represent this process through inference rules or other suitable structures. Going back to our example above, after the father's explanation about how car engines work, the father can infer additional knowledge about the 16-year-old based on the information he explicitly provided. For instance, since his child understands rotational motion, it is likely that they also have knowledge about related concepts such as energy and heat. This understanding is important because combustion, which occurs in car engines, involves the release of heat energy. To simulate this human reasoning process, we employ agents known for their reasoning capabilities. Our idea is to implement these agents in IR context, where the agent is aware of the user's knowledge acquired from the text they have read, as well as any implicit knowledge they may have. To achieve this, we extended the Jason agent programming language, enabling the representation of rules that assist the agent in inferring new knowledge and maintaining representation consistency.

1.2 High-level research questions

We address in this dissertation the following research questions :

— **RQ-I :** In search as learning, what are the effective ways to represent the user's knowledge, learning needs, and their progress towards their learning goals ?

— **RQ-II :** How can we design a system that includes the previous knowledge, the changing user's knowledge and the learning goals as a context ?

— **RQ-III :** How can we evaluate search algorithms that help users in their learning ?

— **RQ-IV :** How can we leverage reasoning techniques to represent and make use of the user's knowledge in search as learning ? In the context of a Belief-Desire-Intention (BDI) agent architecture, how can we include the information retrieval user's knowledge ?

1.3 Thesis contributions

The work presented in this thesis falls under the domain of search as learning, specifically in modeling the user's knowledge and learning needs and evaluating them. We are grateful for the early opportunities to present our initial research questions and directions at two venues. This began with the 11th Italian Information Retrieval Workshop, IIR 2021 (El Zein & da Costa Pereira, 2021b) where we proposed the early stage of our theoretical framework. We presented the developed idea of the framework later at the 23rd International Conference on Principles and Practice of Multi-Agent Systems, PRIMA 2020 (El Zein & da Costa Pereira, 2021a). This presentation marked a significant transition from theoretical aspects to more concrete applications. As the PhD journey progressed, the range and depth of our research broadened, leading to more in depth research questions. This included a significant presentation at the Doctoral Consortium of the 44th European Conference on IR Research, ECIR 2022 (El Zein, 2022). Our participation in these events provided us with valuable feedback that helped refine our research direction at that important milestone and focus our research questions. The contributions of this thesis are highlighted below.

1.3.1 A framework for tracking and estimating user knowledge and needs

We have presented a framework, named `RULK`, that represents the user knowledge and tracks its evolution during the search sessions. We have explored three different approaches for representing the user's knowledge and discussed the advantages and disadvantages of each. Additionally, our framework is capable of estimating the user's learning outcome at any point during a search session.

This work has been conducted as a result of collaborative efforts with researchers from the University of Delft, the Netherlands. The results have been presented at the 3rd International Conference on Design of Experimental Search & Information REtrieval Systems DESIRES 2022 (Câmara, El-Zein, & da Costa-Pereira, 2022), and later at the Conference on Human Information Interaction and Retrieval, CHIIR 2023 (El Zein, Câmara, Da Costa Pereira, & Tettamanzi, 2023).

1.3.2 A novel evaluation metric for documents search results in a learning context

We have proposed a metric that evaluates the relevance of a document with respect to the user's existing knowledge, the knowledge they acquire during the search session, and the information

need they are trying to fulfill. The metric is also capable of evaluating the knowledge gained by a user after reviewing a set of ranked search results. This measure is based on a novel concept of relevance, which we have defined as being relative to each individual user.

This work was initially presented in its preliminary form at the 2nd Joint Conference of the Information Retrieval Communities in Europe, CIRCLE 2022 (El Zein & Pereira, 2022). Subsequently, a complete study with experimental results was presented at the 30th ACM Conference on User Modeling, Adaptation, and Personalization 2022 (El Zein & da Costa Pereira, 2022b).

1.3.3 A benchmark dataset to evaluate retrieval algorithms

We proposed a benchmark dataset that contains a set of documents and their relevance with respect to the potential learning outcomes they could bring. The dataset is used to track the evolution of the knowledge of a set of 500 users when searching to learn about a specific topic. We can trace the knowledge change during the search session at a document level, and this evolution can be compared to other search algorithms to measure their ability to help users learn.

Our work began as a perspective paper, which we presented at the 2nd International Conference on Design of Experimental Search and Information Retrieval Systems, DESIRES 2021 (El Zein, 2021). Subsequently, we shared the dataset and its description in a publication at the Conference on Human Information Interaction and Retrieval, CHIIR 2023 (El Zein & Da Costa Pereira, 2023).

1.3.4 An extension of Jason agent programming language

We extended the Jason programming language agent to be capable of representing the user's knowledge in an information retrieval context. The extended language can also reason about the user's knowledge using rules and maintain consistency in its belief base.

This work has been presented at the International Joint Conference on Web Intelligence and Intelligent Agent Technology WI-IAT 2020 (El Zein & da Costa Pereira, 2020b) and later extended at the International Conference on Agents and Artificial Intelligence ICAART 2022 (El Zein & da Costa Pereira, 2022a).

1.3.5 A comparative study of user behavior in classic search and modern conversational AI

In our ongoing research, we've partnered with the University of Michigan to experimentally compare search behavior and knowledge acquisition between modern conversational AI tools and traditional search methods. To date, our contribution includes the preparation and design of the experiment protocol, as well as the development of a multilayered assessment framework that enables the evaluation of users' knowledge acquisition across all levels of Bloom's taxonomy.

1.4 Thesis outline

The thesis is structured into two main parts. The first part is composed of two chapters : Chapter 2 introducing the basics and evolution of information retrieval, and Chapter 3 discusses the emergence of the Search as learning domain. This section also explores the limitations of traditional information retrieval in addressing complex information needs and reviews literature on adapting search systems to meet learning objectives. The second part of the thesis presents

our contributions to the field of information retrieval for learning purposes. Our first contribution is presented in Chapter 4, where we propose several models to represent the user's knowledge and information needs in IR, including keyword-based, language-based, and entity-based models. Chapter 5 introduces our user-centric evaluation that measures the relevance of documents and rankings while accounting for the user's knowledge and learning objectives. Chapter 6 details a resource dataset we adapted as a benchmark for tracking knowledge gain in search sessions, providing a baseline for comparing retrieval algorithms. Chapter 7 discusses our extension of the Jason agent programming language, enhancing its ability to track and infer user knowledge in the field of information retrieval. Finally, Chapter 8 discusses the overall results of our rseearch as well as future work.

Background and State of the Art

CHAPTER 2

From Classic to Personalized Information Retrieval : Models and Evaluation

This chapter provides an overview of information retrieval, which is the process of retrieving relevant information to satisfy a user's information need, often expressed as a query of terms. Given the exponential growth of information available on the Web, understanding the user, their interests, and their environment has become increasingly important. This necessity has led to the emergence of contextual IR, which considers the environment surrounding the user, such as the device they are using, the task they are trying to achieve, as well as user-related characteristics. In this chapter, we explore the basics of both classic and contextual IR, including the modeling and utilization of user information for personalization. Our focus extends to personalized IR, a subset of contextual IR centered specifically on the user. Additionally, we will discuss various evaluation measures for these systems.

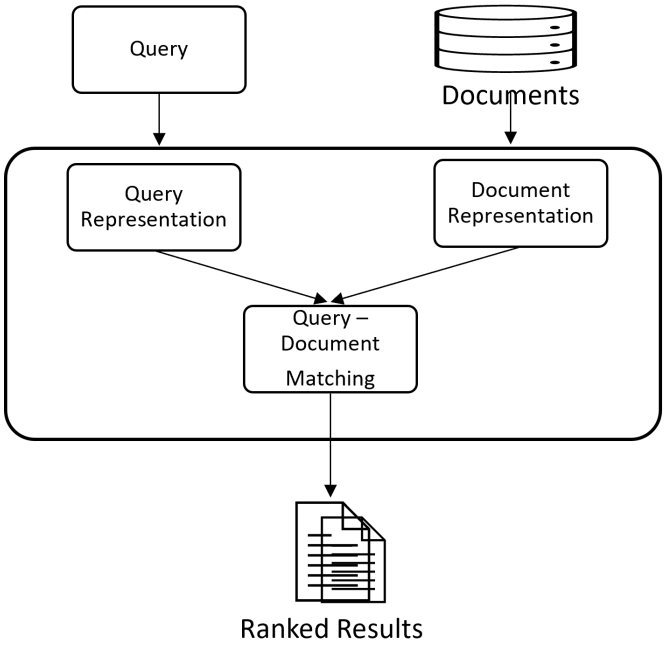

Figure 2.1 – Classic information retrieval model

2.1 Information retrieval basics

2.1.1 Basic notions of information retrieval

Information Retrieval (IR) is, by definition, the process of returning relevant results to a user need, expressed by a query (Baeza-Yates, Ribeiro-Neto, et al., 1999; Schütze, Manning, & Raghavan, 2008). Typically, the retrieved results are ranked in decreasing order of relevance. An IR system searches through a collection of resources, like text documents, web pages, images, or videos, etc. In this thesis, we use *information retrieval systems, search systems, and search engines* interchangeably. Below, we provide definitions for the three core components of this definition :

Document A document is an item that contains information that can potentially satisfy a user's need. While it is commonly referred to as *text*, a document can also be in the form of an image, video, or other media. In the remainder of this thesis, we will use the term *document* to refer to a web page or a search result.

Query A query is a set of keywords expressing a user's need for information.

Relevance A document relevance is the degree to which a document matches the information needs of a user, as expressed by their query or search request (Saracevic, 1970).

2.1.2 Classic retrieval models

A *retrieval model* is a mathematical representation of the notion of relevance. It is the main process behind *matching* and ranking a subset of documents, denoted as d, to a user's need, denoted

as q. We discuss in this section three main retrieval models : the *Boolean* model, the *vector space model*, and the *probabilistic model*.

2.1.2.1 Boolean model

Documents and queries in the Boolean model are represented as sets of keywords. A document is a logical conjunction of keywords. A query is a Boolean expression of terms connected using *Boole*'s basic logic operators (Boole, 1847) : *AND, OR,* and *NOT* which respectively correspond to the intersection of two sets, their union, and the negation of one set.

The notion of relevance in *Boolean models* is simple : a query keyword is either present in the document or not. Consequently, documents are either relevant or irrelevant. The relevance function for this model is then expressed : $Rel(d, q) \in \{0, 1\}$.

2.1.2.2 Vector space model

The vector space model (Salton, Wong, & Yang, 1975) uses vectors to represent queries and documents. These vectors, denoted as $\vec{d}$ and $\vec{q}$ respectively, contain weights and are positioned in a multidimensional space. The size of dimensions in this space corresponds to the number of unique terms found in the document collection, also known as the index terms. Each weight is a measure of the importance of an index term in a document or a query, respectively.

The distance between the vectors denotes the relevance of a document to the query. By calculating distances between all documents and the query, this model generates a ranked list of documents based on their similarity to the query. Cosine similarity is one common similarity measure that calculates the angle formed by the two vectors (Croft, Metzler, & Strohman, 2010). The related relevance function is then : $Rel(d, q) = cos(\vec{d}, \vec{q})$. The more similar the vectors, the smaller the angle between them, and the higher the corresponding cosine similarity value. The system returns the documents in decreasing order of cosine similarity value.

2.1.2.3 Probabilistic model

The probabilistic retrieval model is based on the *Probability Ranking Principle* (PRP), which states that an information retrieval system should rank the documents based on their probability of relevance to the query, given all the available evidence (Belkin & Croft, 1992). The principle takes into account the uncertainty in the representation of the information needed and the documents. The model estimates the probability that a document is relevant to a query.

Briefly, the PRP proposes to retrieve the documents having a higher likelihood be relevant R than non-relevant $\bar{R}$: $P(R/d) > P(\bar{R}/d)$. The relevance score is then calculated as follow : $Rel(d, q) = \frac{P(R/d)}{P(\bar{R}/d)}$.

One of the several work extending this model is the *Okapi BM25* (Robertson & Jones, 1976) that combines term frequencies, inverse document frequencies, and document lengths. The IR community has benefited greatly from BM25, which is still widely used today and considered a solid baseline.

2.1.3 Classic evaluation measures

Information retrieval systems or algorithms are typically evaluated using two key metrics : efficiency and effectiveness. Efficiency focuses on performance factors related to query or re-

quest processing, including response time, processing speed, and resource utilization. On the other hand, effectiveness measures the accuracy and relevance of the retrieved document, taking into account factors such as precision, recall, and relevance ranking. Effectiveness can also be defined by the ease with which users can fulfill their information-seeking tasks and satisfy their information needs. While both efficiency and effectiveness are important for evaluating information retrieval system performance, this thesis will specifically focus on effectiveness.

Classic information retrieval evaluates the individual performance of each query according to the number of returned relevant documents and non-relevant ones. Consequently, the primary challenge of retrieval systems is to maximize the number of relevant documents returned and minimize the non-relevant ones. We present the different evaluation measures used in classic information retrieval in the following.

2.1.3.1 Precision and recall based measures

Precision The *precision* is the number of retrieved relevant documents over the total number of retrieved documents. It is defined as follows :

$$Precision = \frac{relevant\ documents\ retrieved}{retrieved\ documents} \tag{2.1}$$

Recall The recall is the number of relevant documents that are retrieved over the total number of known relevant documents in the document collection. It is defined as follows :

$$Recall = \frac{relevant\ documents\ retrieved}{relevant\ documents} \tag{2.2}$$

2.1.3.2 Rank oriented measures

Precision@K is the fraction of retrieved relevant documents within the top K retrieved documents over the total number of retrieved documents.

Recall@K is the fraction of retrieved relevant documents within the top K documents over the total number of relevant documents in the document collection.

Mean average precision MAP@K is the average precision value at K after each relevant document has been retrieved for a query q. It sums the precision at K for all queries in the set and then divides the result by the total number of queries. At a rank r, the MAP@k is defined as follows :

$$\text{MAP@K} = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{K} \sum_{k=1}^{K} \text{Precision@k}(q) \cdot \text{Rel}_{r,q} \tag{2.3}$$

where Q is the set of queries, $|Q|$ is the number of queries, Precision@k(q) is the precision at rank k for query q. $Rel_{r,q}$ is an indicator function that is equal to 1 if the r-th retrieved document for query q is relevant, and 0 otherwise.

Cumulative gain CG measures the cumulation of gains of all relevant documents up to a certain rank k (CG@k) (Järvelin & Kekäläinen, 2002, 2017). The measure relies on a relevance score *rel* associated with each document, which gets accumulated over the ranking. The cumulative gain at rank k is defined as follows :

$$\text{CG@K} = \sum_{k=1}^{K} \text{Rel@}k \tag{2.4}$$

where $Rel@k$ denotes the relevance score of a document at rank k.

The *discounted cumulative gain (DCG)* is a developed version of *CG* that accounts for the rank of the document in the list. As the user goes lower on a document on the list, the related gain gets discounted by the rank of that document. The relevance score is then divided by the log of its rank. The *normalized discounted cumulative gain (nDCG)* accounts for the ideal *iDCG* listing in a set of documents for a given query. The *DCG* is then normalized by dividing it by the ideal score *iDCG*.

2.1.3.3 Laboratory-based evaluation : The Cranfield paradigm

Laboratory-based evaluations involve controlled experiments where participants are presented with predefined tasks designed to measure a specific retrieval algorithm. They can also be referred to as batch evaluation or test collection evaluation. The *Cranfield* paradigm is a widely recognized laboratory-based model to evaluate the performance of information retrieval systems. It was initiated in the 1960s by Cleverdon (Cleverdon, 1967) from Cranfield University in England, having the following goal :

> A laboratory type situation where, freed as far as possible from the contamination of operational variables, the performance of index languages could be considered in isolation.

The Cranfield paradigm is characterized by a "test collection" having three main components :

1. The corpus set : A large collection of documents that serve as the basis for the information retrieval evaluation. This collection typically includes a diverse set of documents that cover a wide range of topics.
2. The query set : A set of queries representing the information needs of users.
3. Relevance judgments : A set of judgments of the relevance of each document in the document collection to each query in the query set. Those judgments are typically manually performed by human assessors or domain experts who evaluate the documents based on their relevance to the information needs expressed in the queries.

Voorhees outlined three major simplifying assumptions of the Cranfield paradigm (Voorhees, 2002). The first assumes that all relevant documents are equally helpful and that the relevance of one document is independent of the relevance of any other document; this is referred to as *topical similarity*. Furthermore, a majority of experiments assumed relevance as a binary concept. Second, the reliance on expert judgment is assumed to be generalizable for all search users : A document judged relevant by an expert is considered to be relevant for all users. Finally, it is assumed that all relevant documents for the query are found in the corpus. Despite being an old methodology and facing criticism, the Cranfield paradigm is still widely used today, particularly in IR benchmarking and evaluation forums such as CLEF (Conference and Labs of the Evaluation Forum) (Peters & Ferro, 2014) and TREC (Text REtrieval Conference) (Harman, 1996).

2.2 From classic to personalized information retrieval

Classical information retrieval models have proposed numerous methods to represent documents and queries, and match them. However, these models often treat the search environment as isolated, considering only the query and documents while ignoring the user, their preferences, and their search context. This classic approach is flawed as it views the query as the sole representative of the user's information needs, leading to generic search results that do not consider the user's profile. Suppose that two users, Alice and Bob, issue the same query "tax declaration" on a search engine. Alice lives in New York, while Bob lives in Los Angeles. Classic IR models would return the same set of documents to both users, regardless of their geographic location. In a modern approach, we would expect that search engines would take into account the geographic location of the users and adapt the results that are more relevant to their location.

As the amount of available information is generated and accessed by millions of individuals with diverse backgrounds, knowledge, and preferences, the need for results adaptation and personalization becomes increasingly pressing. In this section, we will discuss the existing approaches to adapting and personalizing in information retrieval.

2.2.1 Brief about adaptive information retrieval

Adaptive information retrieval aims to enhance search results by using user-system interaction characteristics in addition to the query (Rocchio Jr, 1971). One of the first reasons for the emergence of adaptive IR is the *information explosion*. The rapid increase in the volume of information available on the Web and in the number of users became a significant challenge for classic information retrieval. A naive matching of a query and documents would return an enormous list of results to the user, causing information overload. Another motivation for adaptive IR was that query terms were often ambiguous or insufficient to find what the user was looking for, particularly when users were unfamiliar with the searched topic and lacked the necessary vocabulary.

Popular methods in adaptive IR include query reformulation (Rocchio Jr, 1971), query disambiguation (S. Liu, Yu, & Meng, 2005), and search result clustering by topic (Navigli & Crisafulli, 2010; Zeng, He, Chen, Ma, & Ma, 2004). Query reformulation proposes new queries by adding new keywords extracted from the search results deemed relevant by the user or using the first few returned results. Query disambiguation techniques assist users in expressing their needs and extending the language used in the query. Clustering techniques on the other hand group the documents in a collection into classes in a way that documents of similar terms, or identified topics, are associated together. The index terms are generally used to determine the document cluster (Tetali, Bose, & Arif, 2013; Zamir & Grouper, s. d.).

However, despite these methods, adaptive information retrieval techniques still exclude the user from the search environment. The relevance of a document is still determined by only two elements : the query and the document. Even though query reformulation and disambiguation techniques serve to better express the user's needs, they do not always help in understanding the user's search goal, motivation, or task they are trying to accomplish. Additionally, these techniques often require explicit feedback from the user on the proposed queries or on the relevance of the documents, which can disrupt the user experience. Therefore, there is a need for a contextual approach to information retrieval that considers not only the query and the documents but also the user's context, preferences, and goals.

2.2.2 Brief about contextual information retrieval

Contextual Information Retrieval emerged as a solution to overcome the limitations of adaptive information retrieval models, which focused only on the query and the user's feedback to enhance search results. Adaptive IR techniques were not enough to understand the user's search goal, motivation, and search context. Contextual IR, on the other hand, is a multidimensional concept that aims to incorporate various contextual factors that can be classified broadly as the user, search environment, and search system. It is built on the premise that search is a dynamic and interactive process influenced by various factors. Contextual information retrieval combines knowledge about the user query and context to answer the user's information needs (Allan et al., 2003). Today's search engines commonly consider the user's geographic location, interests, and search task as important context elements.

Initiatives emerged in 2005 to promote research in the field of contextual IR, with the organization of the ACM SIGIR 2005 Workshop on Information Retrieval in Context (Ingwersen & Järvelin, 2005). This was followed by the Information Interaction in Context Symposium (IIIX) (*IIiX : Proceedings of the 1st International Conference on Information Interaction in Context*, 2006) in 2006 and the Conference on Human Information Interaction and Retrieval (CHIIR) (*CHIIR '16 : Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*, 2016) in 2016. These initiatives have made significant contributions towards enhancing the understanding and development of Contextual IR techniques, enabling the provision of more effective and personalized search results to meet the user's needs. We will provide a brief discussion of various contextual taxonomies and their components. Later, we will focus on the user and task-related context that affects information retrieval and the relevance of documents that we will use later in our contributions.

There have been several proposed definitions of the context within different taxonomies, each with varying features and dimensions. *Cool*'s taxonomy (Cool & Spink, 2002) for example, introduced the search task and the geo-spacial dimension as dimensions of their taxonomy. The mentioned taxonomy included four levels defined as follows : (1) A search environment that includes cognitive, social, and professional factors, (2) the user background, search goals, and intentions, (3) user-system interaction that involves the impact of the environment on the user relevance assessments (4) the query linguistic level. In 2010, Tamine et al. proposed a context taxonomy (Tamine-Lechani, Boughanem, & Daoud, 2010) and defined the context as follows :

> In IR applications, context refers back to the whole data, metadata, applications and cognitive structures embedded in situations of retrieval or information seeking. In particular, those data having an impact on the user's behavior and perception of relevance.

The proposed taxonomy in (Tamine-Lechani et al., 2010) comprises five dimensions : device, spatio-temporal, user, task, and document. The device dimension deals with the search environment, the devices used, or the network characteristics. The user context has two main components : personal context including demographic, psychological, and cognitive context, and social context. The search task dimension deals with two significant aspects : the type of user information required behind the query, such as informational, navigational, or transactional, and the domain of user interest specific to the search task. Finally, the document context dimension is defined based on three sub-dimensions : (1) the document representation, such as structural elements, citations, and metadata, (2) the data source characteristics such as credibility, and (3) the quality of the informa-

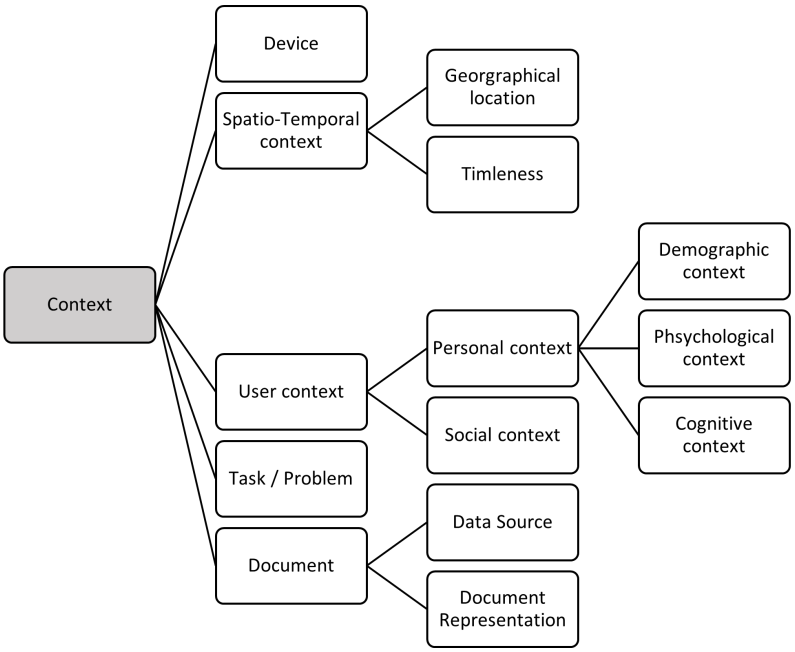

Figure 2.2 – Contextual dimensions in information retrieval by Tamine et al., 2010

tion, such as freshness, precision, coherence, security, and so on. Figure 2.2 shows an illustration of these dimensions.

Out of the five dimensions, we brief the user context that includes four dimensions. Firstly, the demographic context includes user preferences related to their language, location, or gender, as noted by previous studies (Frias-Martinez, Chen, Macredie, & Liu, 2007; Hupfer & Detlor, 2006). Secondly, the psychological context considers the user's emotional state, including anxiety or frustration, which can significantly impact their judgment on search results (Bilal, 2000; K.-S. Kim, 2008). Thirdly, the cognitive context refers to the user's level of expertise and interests, including short-term (Agichtein, Brill, & Dumais, 2006) and long-term interests (Sieg, Mobasher, & Burke, 2004), as well as their familiarity with the search topic and previous knowledge about it. Lastly, the social context is influenced by the user's community, such as friends, neighbors, and colleagues.

2.2.3 Personalized information retrieval

Personalized Information Retrieval (PIR), also known as user profile-oriented IR, is a sub-field of contextual information retrieval that places user characteristics at the center of the research process (Sieg, Mobasher, & Burke, 2007; Agichtein et al., 2006). User profiling is the core of search personalization, involving four important steps : collecting user-related information, representing the profile, updating it (Van Leekwijck & Kerre, 1999), and using it (Gauch, Speretta, Chandramouli, & Micarelli, 2007). Determining the appropriate structure for a user profile can be a difficult task because of the possible changes over time, especially where no standardized evaluation framework exists.Therefore, an accurate representation of the user profile is important to appropriately obtain better retrieval results (de Campos, Fernández-Luna, Huete, & Vicente-López, 2013).

2.2.3.1 User information collection

The first step of user profiling collects information about individual users. This information is usually collected from the user's queries (Bilenko, White, Richardson, & Murray, 2008), the results snippets, and less commonly the content of previously read documents (Matthijs & Radlinski, 2011 ; Biancalana, Micarelli, & Squarcella, 2008 ; Bilenko & White, 2008). The information can be collected on the client side - on the user's machine - or server side. Information about users is gathered explicitly and implicitly or a combination of them (Psarras & Jose, 2006).

Explicit user feedback Explicit approaches rely on users providing the system with additional information about themselves, their interests, and information needs. Some user-related information can be collected during on-boarding, such as demographics including gender, birth date, marital status, education level, or job. Explicit feedback can also be obtained by asking users whether they consider a result relevant or not, which can be collected during or after the search session. One example of explicit feedback is a text box or a short survey that allows the user to express their opinion on a search result. The drawback of these methods is that they depend on the user's willingness to provide personal information or to share their opinion. The accuracy of the user's profile might be affected by the accuracy of the provided answers. Also, in case the user abstained from sharing their information for privacy concerns, lack of time, reluctance, or any other reason, no profile can be built. It is usually not recommended to rely solely on information from explicit feedback, especially those provided during on-boarding, since they can change over time and the user may not remember to update them. This can lead to an inaccurate user profile over time.

Implicit user feedback Implicit approaches rely on information collected about the user without additional interaction with the system. They predominate explicit approaches (Matthijs & Radlinski, 2011 ; Vu, Nguyen, Johnson, Song, & Willis, 2017 ; Cai, Liang, & De Rijke, 2014) and often include browsing history is a common source of implicit information as it contains the user queries, the address of the pages they visited with their related dates and times. This information is easy to collect even if the related designs do not ask the user to provide it. Although the use of these methods has been shown to have more benefits than drawbacks, they can introduce noise that may affect the identification of relevant features. Noise can arise from a link or document being selected by mistake, or if the user spent more time because they got distracted.

2.2.3.2 User profile creation

The user's interest is a frequently represented component of their profile (Park, 1994). Creating the user's profile, often referred to as "modeling" the user, involves organizing their information into a structured format. We present in this section several methods used for interest representation. While not all of these methods are widely used, we will present and discuss the advantages and disadvantages of each one.

Terms-based vectors Vector-based user models are the most common method for representing the user profile. A vector-based model is a feature vector(s) of two components : (1) terms extracted from documents the user marked as relevant and (2) associated weights calculated by a weighting method like term-frequency (TF), TF.IDF (Salton & Yang, 1973 ; Lieberman, 1997), BM25, or probabilistic approaches (Teevan, Dumais, & Horvitz, 2005). The weights reflect the

importance of each term in the profile. The simplest representation is *keywords-based vectors* where the terms are keywords without semantics or reference to ontologies. Another simple, yet common structure of this model is a uni-dimensional vector, where one vector represents the general interest of the user.

Multi-dimensional approaches can represent different aspects of the user's interest or even several interests (Mc Gowan, 2003). Multiple vectors can also represent multiple interests, for example, short-term vs. long-term interests. A user model can be short-term, long-term, or both. The short-term model represents the interests of the user's current research activities, while the long-term model represents the persistent interests of the user and is captured from their entire search history. Grouping search activities by search goal is a common mechanism in short-term modeling. When modeling the user's interests and preferences, it is not always common to save the short-term model : it is built based on the current behavior and immediately used to personalize the results of the current session. Research has shown that collecting more information about users can lead to a better understanding of their interests and improve the relevance of personalization. For example, in (Eickhoff, Collins-Thompson, Bennett, & Dumais, 2013), combining short-term and long-term user profiles resulted in better personalization performance.

The simplicity of vector-based models can be both advantageous and inconvenient. While their implementation is straightforward, maintaining a coherent structure that represents multiple interests at different levels of generality and specificity can be challenging. Additionally, the weights in such models are often calculated based on frequencies and a bag-of-words approach, which may lead to ambiguities in interpretation.

Connected term networks Connected term networks refer to a method of representing a user's interests by associating semantically similar terms with the same node. This approach helps to handle the polysemy of keywords in a user's profile. Nodes and links in the network can be assigned weights to indicate their importance. One advantage of connected term networks over vector-based models is their ability to model the relationship and semantic correlation between a key term or concept and its associated terms.

To construct a user profile using connected term networks, terms are extracted and integrated into a network of nodes. The specific methodology for extracting semantic correlations between nodes can vary between models. For example, IfWeb (Begg, Gnocato, & Moore, 1993) links nodes corresponding to co-occurring terms in the same document. The Wifs filtering system (Micarelli & Sciarrone, 2004) creates network nodes using a pre-constructed database by domain experts and links them based on their co-occurrence. Another approach adopted in (Koutrika & Ioannidis, 2005) connects keywords in the network using logical operators such as conjunction, disjunction, substitution, and negation. Kim and Chan (H. R. Kim & Chan, 2003) classify terms and create a network of hierarchical terms as showed in Figure 2.3.

Connected term networks enrich the representation of user interests by capturing semantics and relations between terms. However, one disadvantage of this model is that it may only contain terms that the user is already familiar with, making it difficult to match with "new" terms when a new need arises.

Semantic and conceptual networks Semantic networks and concept-based representations both aim to represent a user's profile as a network of nodes and relations between them. In semantic networks, the nodes are weighted keywords, and the links between them are based on

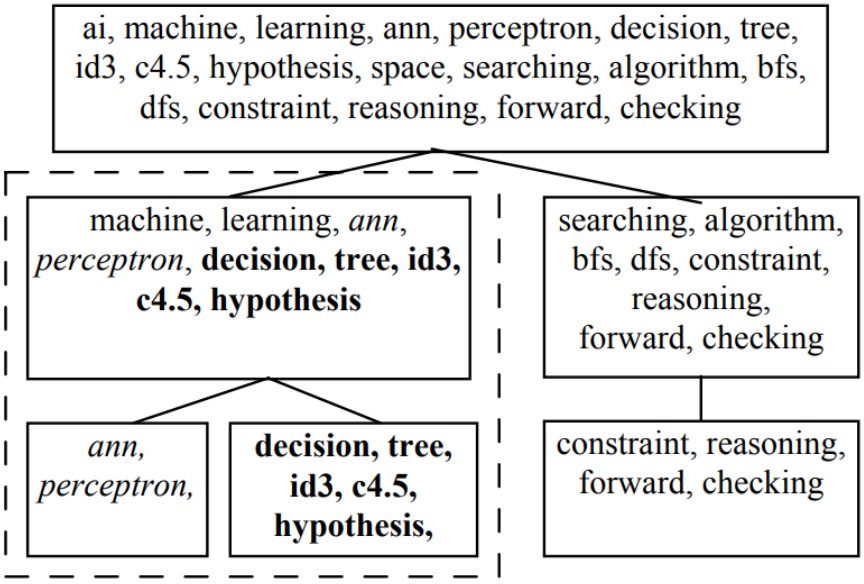

Figure 2.3 – Sample user interest term hierarchy by Kim and Chan, 2003

predefined semantic resources or concept hierarchies. On the other hand, in concept-based representations, nodes represent abstract topics of interest, and the relations between them must respect the topology of the conceptual source, such as a taxonomy or reference ontology.

Both types of representations have a similar structure and can be used to create a user's profile. In semantic networks, high-level concepts in the hierarchy usually represent the long-term profile, while low-level concepts represent a high level of specificity in the short-term user profile. Similarly, in concept-based representations, the higher levels of the hierarchy often represent long-term interests, while the lower levels show more specific concepts related to short-term interests.

One example of a simple hierarchy model used in semantic networks was proposed by Gauch *et al.* (Gauch et al., 2007), which uses a simple ontology with one relation between its parent and child nodes : "is-a" or "has-a". Another approach to concept-based representations is to use a thesaurus like WordNet to map between terms and related concepts. In both cases, the goal is to create a network that represents a user's interests and enables personalized queries.

The utilization of this approach is advantageous as it standardizes the detection of terms and their linking with unified relations. Nonetheless, it should be acknowledged that this technique may encounter sparsity issues when applied to large-scale representations such as the Web, thereby presenting limitations in identifying relevant concepts for a given search among a vast mass of ontology concepts represented in the user profile. As a result, it may significantly increase the execution time of personalized queries and the management of the user profile's evolution.

Daoud *et al.* constructed a weighted graph to represent a short-term user interest during a particular search session (Daoud, Lechani, & Boughanem, 2009). A query profile was constructed using the relevant documents evaluated by the user for each query submitted. The user profile is modeled as a graph consisting of semantically associated concepts derived from the ODP ontology. The graph architecture is showed in Figure 2.4 is a tree-like structure composed of components linked by three types of relationships :

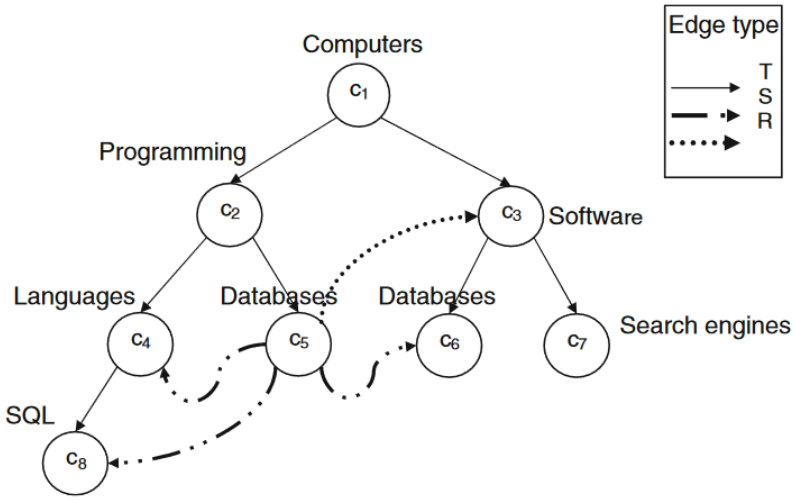

Figure 2.4 – A user profile in a graph-based representation by *Daoud et al.*

— T corresponds to the hierarchical component of the user profile, which is composed of "is-a" connections.
— S corresponds to the non-hierarchical component of the user profile, which is composed of "symbolic" cross-connections.
— R corresponds to the non-hierarchical component of the user profile, which is composed of "related" cross-connections.

Deep learning representations More recently, representations based on deep learning have been investigated to build fine-grained models of users (Zhou, Dou, & Wen, 2020). For example, Vu et al. represented a user by a user embedding that is learned based on the user's previous queries and clicked documents at the session level (Vu, Willis, Tran, & Song, 2015). Ge *et al* generated user profiles based on recurrent neural networks from the user's long-term interests (Ge, Dou, Jiang, Nie, & Wen, 2018). A query-aware attention model is also constructed based on the user's current session and used to weigh the long-term interests. Lu *et al.* applied generative adversarial networks to generate discriminative negative examples to build fine-grained user models (Lu, Dou, Jun, Nie, & Wen, 2019).

One of the primary drawbacks of using deep learning-based representations to build user models is that they can be non-human readable and lack expressiveness. This can make it difficult for humans to interpret and understand the models, especially when it comes to understanding how the model arrived at its decisions.

2.2.3.3 User profile update

Updating the user profile is the process that manages its evolution with the arrival of new information. It is a complementary process to the profile construction and is an incremental process that adds new information to the representation of the profile.

As discussed in the previous section, a user profile can incorporate weights, such as weighted terms or graphs, which can be adjusted during the update process. To represent the changes in the user's interests, the profiles might be reweighted to increase or decrease (Ge et al., 2018). Previous research has also taken into account the aging or decay of user interests (Stefani & Strapparava, 1998; Asnicar & Tasso, 1997), and the weights are updated accordingly.

For representations based on vectors, some studies have proposed a limit on the number of vectors representing the user's interests (L. Chen & Sycara, 1998). When this limit is reached, the two most similar vectors are merged by retaining only the top M-weighted terms, based on cosine similarity. This method can lead to the clustering of terms that appear in multiple interest vectors.

Micarelli and Sciarone proposed a user modeling system called HUMOS (Micarelli & Sciarrone, 2004), which utilized the Justification-based Truth Maintenance System (JTMS) to ensure the consistency of the user model. HUMOS represented long-term models of individuals using frames containing informative words and semantic networks. To prevent contradictions in the model, logical constraints based on justification were imposed by the TMS during updates. For example, if a user removed an active stereotype, all assertions that were justified by it would also be removed to maintain consistency in the model.

2.2.3.4 User profile exploitation

After capturing and organizing user information into a user profile, the next step is to leverage this profile to enhance search performance. The objective is to utilize the profile to better under the user and their intention and therefore propose more relevant documents.

Query expansion Query expansion is the process of reformulating or modifying - also said "augmenting" - a given query to retrieve more relevant results (Manning, 2008). This process typically uses the information - terms - in the user's profile model to expand the initial query. This process is sometimes needed because users do not always have the representative vocabulary to formulate their queries and express their needs (Furnas, Landauer, Gomez, & Dumais, 1987); the resulting queries might be ambiguous (Buttcher, Clarke, & Cormack, 2016). The process can also involve modifying the query terms' weights. Query adaptation can happen automatically or semi-automatically. A detailed survey about personalization techniques can be found in the work of Ghorab *et al.* (Ghorab, Zhou, O'connor, & Wade, 2013).

Result adaptation In most search systems, search results are typically presented to the user in the form of a ranked list of results. Adapting this list can be performed by three different methods : (1) result re-ranking, (2) result filtering, and (3) result scoring.

Result filtering is essentially a refinement of result re-ranking. In result filtering, the list of results is first sorted in descending order of relevance scores, and then any results that do not meet a certain threshold are removed from the list and not presented to the user.

Result re-ranking reorders the results using a pre-settled relevance criterion that pushes the documents matching the user's profile to the top ranks (Chirita, Olmedilla, & Nejdl, 2004; H. Liu & Hoeber, 2011; You & Hwang, 2007). Re-ranking methods are wrapped around an original retrieval system or a well-known search engine. A common practice is to re-rank the top N results

rather than the entire list. The approaches to results re-ranking are aimed at modifying the ranking score by explicitly matching the user profile against the user query and then combining the obtained score with the relevance-based score produced by the traditional IRS or search engine. Re-ranking techniques proposed in the literature may differ both in the adopted user model and in the re-ranking strategy. For example, in the *MiSearch* system (Speretta & Gauch, 2005a), the concepts in the documents were detected using text classification techniques and then compared to the ones in the user profile using cosine similarity. The results were re-ranked in decreasing order according to the similarity between the user profile and the document snippet. Knowing that almost all re-ranking algorithms do not involve user interaction, we cite some work that refined the re-ranking based on the user's feedback on the top n documents (Tanudjaja & Mui, 2002; C. Yu, Liu, Meng, Wu, & Rishe, 2002).

Result scoring, on the other hand, involves including adaptation features directly into the primary scoring function of the system's retrieval component. Unlike search re-rank and filtering, search scoring happens only in one round. It evaluates the document's relevance in a single step, without intermediate stages or user profiles. The relevance judgments are directly employed in ranking systems. Agichtein et al. (Agichtein et al., 2006), for example, employed a machine learning model to teach the system to assess the document's relevance based on a set of adaptive factors (also referred to as implicit features) such as query length or time spent on each page.

2.3 Personalized IR evaluation

Traditional evaluation measures discussed in Section 2.1.3 are suited for classical information retrieval models, where the search environment only considered the query and the document. They were constructed to accurately return documents that match the user's queries and were mainly evaluated using precision and recall. The emergence of contextual and personalized information retrieval has highlighted the necessity to develop novel evaluation metrics that account for the dimensions and metadata used by these systems (Pasi, 2010).

We will focus on measures and approaches for personalized systems based on the user's context. These measures, as identified by Kelly's taxonomy (Kelly et al., 2009) for interactive IR, provide valuable insights into the characteristics and knowledge of the searcher, which can significantly impact their information-seeking behavior and performance. The user's context is one of the four measures defined by Kelly's taxonomy, and it includes factors such as the user's age, sex, prior search experience, and knowledge of the topic. The other three dimensions in Kelly's taxonomy are interactions, performance factors, and usability measures, which respectively focus on the searcher's interactions with the system, capture the outcome of the searcher's interactions, and assess the user's perception and satisfaction with the system.

The evaluation of the retrieval system can also be categorized into two other dimensions : *system-based measures* and *user-based measures*. System-based evaluation involves evaluating the effectiveness and efficiency of the IR system as a whole, without considering individual user preferences or behavior. On the other hand, user-based evaluation in IR considers the perspectives and feedback of users who interact with the IR system.

In this section, we present an overview of the main evaluation approaches proposed in contextual and personalized IR, with a focus on the user and task context.

2.3.1 A need to revisit the notion of relevance

The concept of relevance is fundamental to IR evaluation as it determines the criteria used to rank and present documents to users. The traditional notion of relevance is based on the similarity between the query and document representations, and it is often addressed in isolation and only at the topical level, such as the matching of topics between the documents and the user's query (Huang & Soergel, 2013). Additional relevance dimensions have been introduced in the literature, such as coverage, which measures the degree to which the user's interests are covered in a document (Pasi, Bordogna, & Villa, 2007) ; appropriateness, which evaluates the document's suitability with respect to the user's interests (da Costa Pereira, Dragoni, & Pasi, 2009) ; novelty, which measures how unique the document is with respect to what has already been suggested to the user by the system (Clarke et al., 2008) ; and diversity (Agrawal, Gollapudi, Halverson, & Ieong, 2009 ; H. Chen & Karger, 2006).

The emergence of contextual and personalized information retrieval urged the redefinition of the traditional notion of relevance : relevance should go beyond the user's query and the set of documents to include the user's context and preferences. In other words, the relevance of information should be determined not only by its textual similarity to the user's query or the popularity of the document, but also by how well it matches the search context including the user (ie. needs, interests, and goals) and their surrounding context (Borlund, 2003b).

Since the context of the search was introduced in the retrieval algorithms, they had to be included in a new definition of relevance and evaluation measures. Therefore a high relevance between the document and query may not necessarily mean the document is useful for the user (Mao et al., 2016). Saracevic (Saracevic, 1975) defined the relevance as follows :

> Relevance has a context, external and internal . . . Context : the intention in the expression of relevance always from context is directed toward context. Relevance cannot be considered without context.

The relevance of a document is subjective to each user, and the question asked when evaluating a system must change from answering the question *"Is the system able to select relevant documents ?"* answer the following question "How accurate is the system in retrieving documents that are relevant to a specific user and their surrounding context" (Tamine & Daoud, 2018). The need to revisit the notion of relevance has led to the emergence of new evaluation approaches for personalization systems.

2.3.2 Evaluation approaches and measures in PIR

In this section, we aim to provide an overview of the major methods used to assess the effectiveness of personalized information retrieval systems, while following the approach proposed by Tamine *et al.* (Tamine-Lechani et al., 2010 ; Tamine & Daoud, 2018) for organizing these methods hierarchically. Figure 2.5 presents our proposed categorization. Specifically, we categorize the assessment approaches into two main groups : retrieval effectiveness, which further divides into system and user approaches, and profile accuracy on the other side.

2.3.2.1 Retrieval effectiveness

Information retrieval effectiveness refers to the ability of an IR system to accurately retrieve relevant information in response to a user's information-seeking tasks or queries. The effectiveness

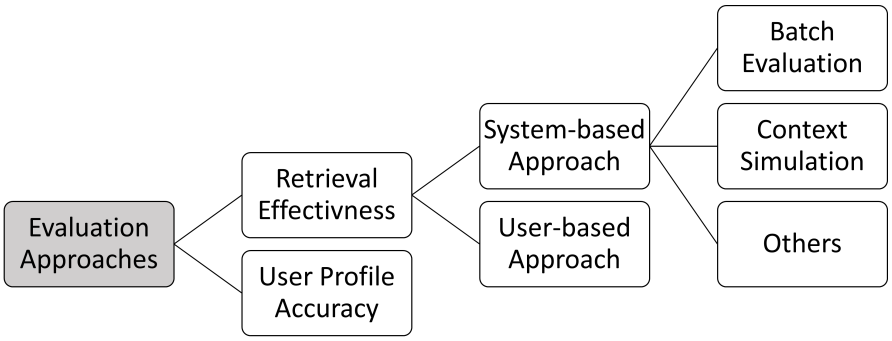

Figure 2.5 – Categorization of personalized IR evaluation approaches

of a retrieval system can be measured through two distinct approaches : a user-based approach and a system-based approach. In the user-based approach, effectiveness is evaluated by monitoring user behavior and collecting data on how easily and accurately they can complete specific information-seeking tasks. This approach directly assesses the system's ability to fulfill the user's goal of completing a search task, providing valuable insights into the user's experience. On the other hand, in the system-based approach, effectiveness is measured using numeric metrics that score system runs against a set of relevance judgments. This approach is quantitative and repeatable, allowing for objective evaluation of the system's performance in relevance and accuracy. Both approaches have their benefits, with the user-based approach providing insights into the actual user experience and the system-based approach offering quantitative and repeatable measurements.

In this subsection, we provide an overview of these two main approaches for measuring, while also discussing how the search context is integrated into these approaches. We briefly highlighted the pros and cons of each approach as well.

System-based approaches

Batch evaluations with contexts TREC is an annual event where researchers from academia and industry come together to evaluate and benchmark IR systems on various tasks like question answering, entity recognition, and ad hoc retrieval. Each track typically involves the development and evaluation of IR systems on specific tasks or challenges.

In batch evaluation, the relevance of documents in the corpus is judged relevant or not by experts or annotators. These relevance judgments are used to evaluate the effectiveness of a retrieval algorithm using a predefined metric. Batch evaluations facilitate rapid experimentation and quick generation of results which allows for several runs of the experiment to fine-tune the parameters and modifications in the retrieval algorithms.

Test collections in batch evaluation are designed to be cost-effective and facilitate comparisons by eliminating potential sources of variability (Bailey, Moffat, Scholer, & Thomas, 2015). The Cranfield and TREC paradigms, for example, exclude variables like users and tasks are removed from the test collection, leaving topics as the main variable. However, these approaches have faced criticism for their limited generalizability of document judgments and for neglecting the notion of relative relevance. Furthermore, the effectiveness of their performance is currently evaluated based

on just one individual query, whereas users typically submit multiple queries, which collectively provide significant insights into the user's intent and goals.

This laboratory-based evaluation was extended by modeling minimal user-system interaction between the user and the system and by including some contextual factors that might affect the relevance judgment of a user. For example, the TREC Contextual Suggestion Track introduced in 2012 (Hubert & Cabanac, 2012) considers user contexts such as location, time, and task to evaluate systems that provide suggestions or recommendations based on the user's context. The TREC Federated Web Search Track (Demeester, Trieschnigg, Nguyen, Zhou, & Hiemstra, 2014) introduced in 2013 involves multiple search engines collaborating to fulfill a user's information needs, and takes into account query contexts such as location or language preference. The TREC News Track (Soboroff, Huang, & Harman, 2018), which has been running since TREC's early years, evaluates news retrieval systems that often need to consider user contexts such as time, language preference, and geographic location. The interactive track (Harter & Hert, 1997) took place between 1995 and 2002 and included some data about the users issuing the queries like the URL of web pages visited by each user. The objective behind this track is to study search as an interactive task and investigate the outcome resulting from a search task. One other noticeable track was the High Accuracy Retrieval from Documents (HARD) (Allan, 2005) that took place for three consecutive years in 2003 and 2003. The query tests in the HARD tracks were annotated with metadata about some user context ranging from biographical data such as sex, age, and spoken language, to information-seeking context such as familiarity of the user with the topic search and the purpose of its search.

While these tracks have incorporated user contextual information in their evaluations, the collected contextual information is generally limited to more or less static aspects, such as location, time, language preference, or task. Dynamic factors like the user's cognitive state, knowledge level, or real-time situational factors are not typically considered in these evaluations. They still rely on classic evaluation measures like recall and precision for their overall evaluation.

Context simulation Context simulation in information retrieval assessment refers to the practice of creating artificial scenarios to evaluate the performance of IR systems (Bouidghaghen et al., 2011). Those scenarios mimic real user-system interactions in IR. They allow researchers to test and compare the performance of IR systems under several conditions and settings, in a timely efficient manner. An evaluation scenario consists usually of a testbed comprising use cases and a fixed set of hypothetical context situations. It is also less costly than running real user experiments, and similar to lab-based evaluations, it is easy to compare between different users and different scenarios. One disadvantage of this type of evaluation study is that it assumes that all searchers in a scenario would interact in the same way.

A simulated context-based evaluation strategy is characterized by the following : users are generalized, the assessors pre-define the search needs (topics), and the relevance judgments. Major evaluation campaigns like TREC, CLEF, and NTCIR often organize evaluation tracks behind such studies.

This approach for evaluation measure is commonly used by researchers testing the system interface design (Mostafa, Mukhopadhyay, & Palakal, 2003 ; White, Ruthven, Jose, & Rijsbergen, 2005). For example, Câmara *et al.* in (Câmara, Maxwell, & Hauff, 2022b) tested several interface designs that scaffolded users in searching for topics and subtopics. They used simulated users with four different profiles : The first one, *Greedy*, follows a subtopic ordering that is optimized for human understanding and attempts to master one subtopic before moving to the next. The

Greedy-Skip user moves to the next subtopic with the next lowest completion value to minimize the number of documents to be read by querying in a domain with lesser knowledge. The *Reverse* user examines the subtopics in reverse order to learn the most complex subtopics before moving to easier ones. The last type, *Random*, is a user that randomly selects a new subtopic after each query with no predefined order, modeling a non-rational learner.

Others The relevance judgment data is not entailed in log files. In most cases inferred through behavioral evidences analysis. There document pair preferences are assumed on the basis of "download," "not download" a user's actions.

User-based approaches User studies in IR evaluations typically involve a set of participants subjects who are given one or more search tasks to perform using a search tool, while their interactions are being logged. The logged interactions can include various aspects such as user queries, visited documents, time spent on each document, mouse, and eye-tracking behavior. Additionally, surveys and interviews before or after the experiment can be employed to collect non-search-related information from users, such as demographic details and their level of expertise in the domain being searched. This additional data can provide valuable insights into the users' characteristics, which can aid in better understanding their behavior and interactions with the search tool.

The main evaluation measures used in user studies are precision@k (Ding & Patra, 2007), NDCG (Agichtein et al., 2006) and the average ranking of search results clicked by users (Speretta & Gauch, 2005b).

One of the key advantages of this approach is that it allows for capturing natural user behavior and interactions of real users, providing valuable insights into how users interact with the system. When the study is well-designed, the logged information can be analyzed to gain rich insights, including feedback about the interface and system effectiveness, as noted by Moffat et al. (Moffat, Thomas, & Scholer, 2013). However, there are also some drawbacks to this approach. Firstly, it can be time-consuming to design the study, recruit participants, conduct experiments, collect data, and analyze the results. Additionally, it may require significant financial resources when compared to simulated and lab-based approaches. Secondly, it could be challenging to replicate the experiments with the same environmental or cognitive setup if any modifications or fine-tuning of the algorithm are needed. This may make it difficult to compare results across different scenarios, unlike some of the other evaluation approaches discussed earlier.

2.3.2.2 User profile accuracy

We discussed in Section 2.2.3 the key steps in PIR, including collecting information, modeling the user with a profile, and personalizing the results according to the created profile. To evaluate the accuracy of the created models to ensure that the represented aspect of the user is reflective and how well the representations captured the aspects. To do so, the representations are compared to the actual aspects. Unlike classic retrieval algorithms, there are no universal measures for the user profile especially since there are many different ways to model the user.

Qiu and Choo modeled the user's preference as a vector of m topics with relative degrees on each topic (F. Qiu & Cho, 2006). The mentioned work measured the *relative error* between the represented profile Te and the reference context T, the difference between the two vectors was calculated as follows :

$$E(te) = \frac{|Te - T|}{|T|} \tag{2.5}$$

Ding *et al.* proposed a self-organizing map to profile users using ontology references (Ding & Patra, 2007). The map was created using a deep neural network classifier trained on labeled documents with their categories. The documents were classified into the categories closest to the user's profile. To evaluate the accuracy of the user profile, the number of correctly classified documents N_{dc} was compared to the total number of labeled documents N_{dt}, as follows :

$$Accuracy = \frac{N_{dc}}{N_{dt}} \tag{2.6}$$

2.4 Conclusion

In this chapter, we presented key concepts in information retrieval and contextual IR. We discussed various user profiling models that can represent a user, specifically their interest. These models can later be used for personalization. We also covered common evaluation methods used to assess personalized search systems. However, it is important to note that there is no universally accepted evaluation method, primarily due to variations in data availability and reproducibility. In subsequent chapters, we will build upon the foundational concepts discussed in this chapter to introduce our own profiling models for representing user knowledge and information needs in Chapter 4, propose our evaluation measure in Chapter 5, and introduce our benchmark dataset in Chapter 6.

CHAPTER 3

Search as Learning

The process of acquiring new information on the Web is often associated with the usage of information retrieval systems, specifically search engines. However, traditional search tools may not be adequately adapted to support users in growing their knowledge and achieving their learning goals, which are expressed as a set of information needs. In this chapter, we explore the concept of search as learning, an iterative process in which learners interact with search tools to acquire knowledge about a specific learning goal. We examine existing research in this interdisciplinary field, identifying challenges and opportunities for utilizing search in this new way. Specifically, we explore the relationship between learning and information retrieval and identify the need for an expressive representation of the user's knowledge state and information need to support effective search as learning.

3.1 Introduction

The process of searching for information has been a primary mean of acquiring knowledge for centuries. Bertram Brookes argued in 1980 that the process of searching for information allows searchers to acquire new knowledge, regardless of the type of search task, simple or complex (Brookes, 1980). Gary Marchionini later described information seeking as "a process, in which humans purposefully engage in order to change their state of knowledge" (Marchionini & Maurer, 1995). Thus, we have known for some time that search is driven by the higher-level human need to gain knowledge. However, when this need is complex, the process of accessing the information can also be complex, involving activities such as query formulation, result assessment, relevance judgment, reading relevant documents, and query reformulation to deepen understanding or find previously unidentified information. The internet and the vast amount of information it contains have made information accessible from anywhere, at any time, and on any device. Search tools, which are also available on every device, have become learning tools that can serve as standalone resources or as complementary materials to other sources such as books, courses, and e-courses. However, this does not mean that finding the needed information for a learning task is easy, as significant effort is required. Users often do not have the proper keywords or vocabulary to express their needs in the form of effective queries, and they may not even know what they do not know. The Anomalous State of Knowledge (ASK) hypothesis (Belkin et al., 1982), proposed by Belkin, suggests that people engage in information seeking when they face uncertainty or confusion due to a knowledge gap on a topic or problem, and information retrieval systems aim to help them resolve this gap. Consequently, a user's knowledge state for a complex learning task may be unlikely to change after a single query or after reading a single document.

Marchionini defined an iterative process where learners intentionally interact with a search engine by reading, scanning, and processing a large number of documents, with the ultimate goal of acquiring knowledge about a specific learning goal (Marchionini, 2006). This process is now commonly known as *search as learning*. We define it as an interactive information retrieval process that considers the user's knowledge state and search needs. Therefore, an "ideal" IR system should help users fill their knowledge gap and consider their current knowledge state and information need.

In this chapter, we will explore the emergence of search as learning as a tool and how learning and information retrieval are related. We will expose existing research on this interdisciplinary field, and examine the challenges and opportunities of utilizing search in this new way.

3.2 Searching as a learning tool

3.2.1 Learning taxonomies

Learning is the cognitive and social process through which knowledge is gained, modified, reinforced, and applied. The objective behind the act of learning is transitioning from one knowledge state to another, for example from a novice to an expert.

The Bloom's taxonomy (Bloom, Engelhart, Furst, Hill, & Krathwohl, 1956; Bloom, 1956) is a framework used in the education field for defining learning objectives and preparing courses and evaluations accordingly. The taxonomy outlines six levels of cognitive skills necessary for learning : remember, understand, apply, analyze, evaluate, and create. The levels are depicted in Figure 3.1, and represent increasing complexities of learning, ranging from fact-based recall to me-

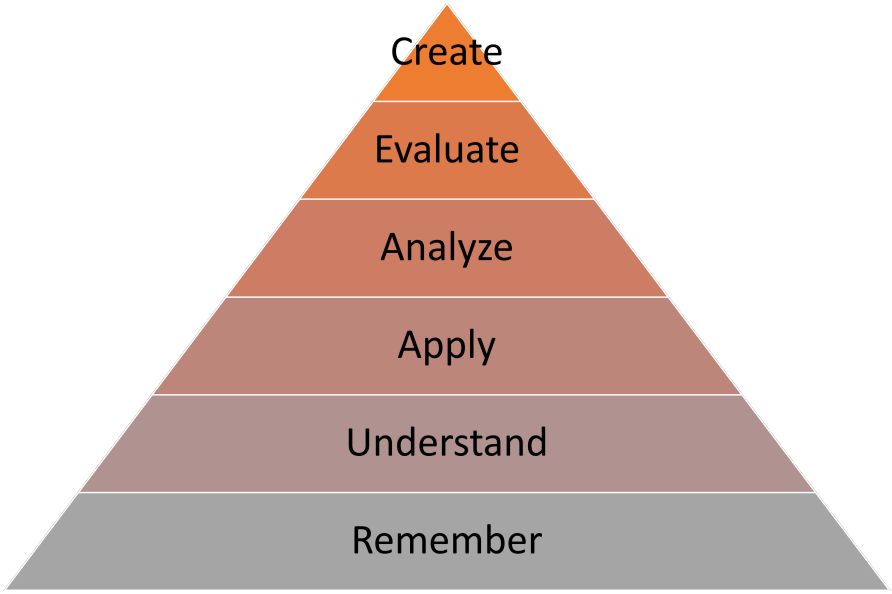

Figure 3.1 – Taxonomy of educational objectives by *Bloom*, 1956

tacognitive skills. Each level involves interaction between new knowledge and existing knowledge structures, with the lowest level, such as *remember*, adding new knowledge to existing structures, the middle levels requiring adaptation of existing knowledge, and the highest levels involving a restructuring of existing knowledge. Over time, Bloom's taxonomy has been extended to include new levels and dimensions (Krathwohl, 2002) ; the original framework however continues to be widely used in education and instructional design. In this thesis, we will follow the original Bloom's taxonomy to discuss learning objectives and outcomes, recognizing its importance in the field of education. Our focus in measuring the learning outcome will be on the lowest level of Bloom's taxonomy, which is "remember."

3.2.2 Web search taxonomies

Search engines have evolved to become incredibly versatile tools, thanks to their access to a vast and diverse pool of information. As users have become more aware of the capabilities of search engines, their usage has also become more diverse. To investigate and classify the user's goal or intention behind their search activity, several taxonomies have been proposed. This is essential in determining what type of information the user is looking for, and providing relevant search results with the appropriate level of detail and presentation.

One such taxonomy was proposed by Broder, who defined three main classes of search activity : navigational, informational, and transactional (Broder, 2002). In the navigational category, the immediate intent is to reach a particular site. In the informational category, the intent is to acquire some information assumed to be present on one or more Web pages. In the transactional category, the intent is to perform some Web-mediated activity.

Other researchers, such as Jansen *et al.* and Rose *et al.* (Jansen, Booth, & Spink, 2007 ; Rose & Levinson, 2004a), have followed similar categorizations with three classes of search activity. Rose *et al.* had a slightly different taxonomy, where the third category was based on the resource instead of the transactional intent.

Marchionini identified three kinds of search activities : lookup, learn and investigate (Marchionini, 2006). Lookup is the most basic kind of search task, and it aims to return discrete and well-structured objects such as numbers, names, short statements, or specific files of text or other media. Lookup tasks are often embedded in learning or investigative activities. Learning searches involve multiple iterations and return sets of objects that require cognitive processing and interpretation. Searching for investigation requires not only several queries and search iterations but also a long period to change the user's personal knowledge base. The mentioned work argues that it is possible for individuals to simultaneously conduct various types of searches, and certain activities may be intertwined with one another. For example, when someone is learning or investigating, they may also perform lookup activities.

In a large-scale study on Microsoft Bing commercial search engine over a two-month period, Russell identified around twenty different search tasks (Russell et al., 2009). "Discover more information about a specific topic" was the second-highest fraction of queries issued per session of nearly 14.5%.

Analyzing the query logs, all the mentioned works reported that informational queries have the highest proportion of search intents, with 49.7% (R. Yu, Gadiraju, & Dietze, 2018) 48%, (Broder, 2002), 63% (Rose & Levinson, 2004a), and 81% (Jansen et al., 2007). These numbers show that search engines are primarily used to find new information and learn new subjects or tasks. Identifying the task that the user is performing during their search is an important aspect to consider when we want to adapt or personalize search results to their specific needs and requirements.

3.2.3 The importance of search tools in learning

While the use of Web search tools for learning is not new, over the past two decades, search engines have become an increasingly important tool for finding information and learning. In the early 2000s, students began to prefer using Web search engines for educational purposes and as an assistant to their learning (Bilal, 2000; Netday, 2004). Research has shown that a majority of students use search engines to begin solving their information needs (Hölscher & Strube, 2000; De Rosa, Cantrell, Hawk, & Wilson, 2006), and more than half of the students claimed to use search engine basis (Griffiths & Brophy, 2005). The number of participants using search engines as a primary source of information has steadily increased over the years, with studies reporting a rise from 19% to 57% between 2005 and 2007 (Dutton & Helsper, 2007) and from 24% to 31% from 2005 to 2009 (Judd & Kennedy, 2010). The latter study also found that Google search was the primary source for information-seeking among university students.

Nowadays, with the excess of information available on the Web and given the advancements of AI tools that could go beyond just assisting teachers, this field is more important than ever. Weller discussed three transformational changes that the use of technology will bring to learning (Weller, 2011) : the dramatic increase in the quantity of open, scholarly information available ; the widespread availability of open networks through which to share and discover information ; and the range and variety of information newly being considered as legitimate examples of scholarly activity. In a recent study, Constantino *et al.* highlighted the importance of a human-centered approach in online teaching and learning, especially in the face of the advancement of different artificial intelligence fields (Constantino & Raffaghelli, 2020). That means that intelligent models should be explainable and adapted to human learning.

3.3 The emergence of search as learning

The search as learning field is founded on idea that learning is a critical outcome of search activities and that search systems can and should be designed and evaluated as tools to support learning. The process of search as learning, which is defined as an iterative process where learners engage with a search system by reading, scanning and processing a large number of documents with the ultimate goal of gaining knowledge about one specific learning objective, was first formally defined by Marchionini (Marchionini, 2006). In the last decade, several summits have been held to develop research agendas in the area of search as learning.

We discuss first the *Strategic Workshop on Information Retrieval in Lorne (SWIRL)* in 2012 (Allan, Croft, Moffat, & Sanderson, 2012) which was one of the early events that emphasized the importance of supporting learning during search. The workshop identified three key directions to pursue, including moving beyond a simple ranked list of results, developing search tools to support learning, and modeling contextual factors that may impact learning during search. The seminar emphasized that complex tasks require exploration, learning, user collaboration, and different information-seeking stages and search strategies.

The *Dagstuhl seminar 13441* on evaluation methodologies in information retrieval in 2013 proposed a research agenda called "from searching to learning" (Agosti, Fuhr, Toms, & Vakkari, 2014). The agenda emphasized the importance of conceptualizing learning as search outcomes and discussed the bottlenecks slowing the advancement of search as learning research. These bottlenecks include the reliance on small-scale lab studies that may lack ecological validity, the lack of awareness of relevant research in other disciplines, and the lack of shared research infrastructure. Additionally, attendees identified four directions to explore in future work, including understanding search as a learning process, understanding how contextual factors can influence learning processes, developing materials to measure learning, and developing search tools to support learning.

Furthermore, several other workshops and special issues in conferences and journals have been devoted exclusively to search as learning. For example, the search as learning workshop Information Interaction in Context Symposium held in 2014 (Freund et al., 2014) and the search as learning workshop at the *Special Interest Group on Information Retrieval (SIGIR)* conference (Gwizdka, Hansen, Hauff, He, & Kando, 2016) in 2016. In addition, there are special issues in journals like the Journal of Information Science (Hansen & Rieh, 2016) and the Information Retrieval Journal (Eickhoff, Gwizdka, Hauff, & He, 2017) that focus on search as learning.

Later, in 2017, the *Dagstuhl Seminar 17092* brought together researchers from different disciplines to discuss four different views on the topic of search as learning, including interactive IR, psychology, education, and system-oriented IR (Collins-Thompson et al., 2017). The motivation for the seminar stemmed from the fact that information retrieval systems are engineered and optimized to fulfill lookup tasks instead of complex search tasks. A complex task requires "exploration and learning, user collaborations, and involve different information seeking stages and search strategies."

The search as learning community, in the aforementioned events, has been exploring and promoting a range of research questions, including factors that influence learning during searches, such as the individual searcher's characteristics, the characteristics of the search task (i.e., the learning objective), and the characteristics of the search system. Moreover, studies have also investigated the relationship between specific behaviors and learning outcomes.

In the following section, we will discuss the advancement of research on these open questions.

General Aspect	Category	Subcategory
User-related	Demographic information	Gender, age, etc.
	Cognitive style	Field-dependent/independent
	Cognitive ability	Perceptual speed, working memory, dyslexia
	Personality Knowledge	Big 5 factor
Situation-related	Search experience	Domain knowledge, topic knowledge
	Task type	Task product, complexity, difficulty, stage, others
	Location & Time	Location, time
	Information object features	FAQs, list format, genre
	Others	Language, health literacy, social network, etc.

TABLE 3.1 – Search contextual factors as presented by Liu *et al.* 2020.

3.4 Understanding user learning and behavior in Web search

One essential step before creating effective learning systems is to study and understand both the external and internal factors that influence the user's learning. External factors, such as the design of the search tool and the type of returned results, can impact how users acquire information. On the other hand, internal factors, such as cognitive and personal characteristics, can also significantly affect the learning progress toward search goals. In this section, we provide a brief overview of current research on search as learning, focusing on the study of these factors.

Table 3.1 illustrates the factors reviewed by Liu *et al.* that influence search behavior (J. Liu, Liu, & Belkin, 2020), encompassing user-related factors such as demographic information, cognitive style, personality, and knowledge, as well as non-user factors that pertain to the situation, such as task, time, and location. These factors are similar to the contextual considerations for personalized search proposed by Tamine *et al.* for search in general (Tamine-Lechani et al., 2010).

3.4.1 The document

The features of a document can significantly impact human learning outcomes. The structure of the page is one such factor that can influence comprehension. Freund *et al.* conducted a study (Freund, Kopak, & O'Brien, 2016) that found plain-text filtered documents to be more effective in improving learning outcomes compared to HTML format. However, other research has suggested that images, when used appropriately, can have a positive association with learning outcomes (Mayer, 1997 ; Verma, Yilmaz, & Craswell, 2016). This suggests that the impact of content on learning outcomes may depend on various factors, such as the presence of advertisements and the proportion of images in the document. In a vocabulary learning task, it has also been found that having a reasonable number of relevant images in a document can improve the learning outcome (Syed & Collins-Thompson, 2017a). In addition to content, the keyword density of a document can also impact learning outcomes. Syed and Collins-Thompson (Syed & Collins-Thompson, 2017a) developed a retrieval algorithm that prioritized documents with a higher density of vocabulary items that users needed to learn. Their experimental results showed that this approach led to higher user learning outcomes compared to a baseline system. Furthermore, embedded links in documents can also impact learning outcomes by disrupting the linearity of the learning process (Zumbach & Mohraz, 2008) and adding additional cognitive load (DeStefano & LeFevre, 2007). As such, educators should be cautious when incorporating embedded links in educational materials. Finally, we

mention some recent work (Hingoro & Nawaz, 2021) studied the rankings of retrieval algorithms in search as learning and proved the popularity and the up-to-dateness of the information source, or the document can impact the user's learning experience too.

3.4.2 Search tool design

User interfaces can be the first point of contact between the user and the search system. Some previous work studied some design features that can help the users in their learning.

Demaree *et al.* found some first indications about the impact of the device used for search in a learning task has an influence on the user's behavior (Demaree, Jarodzka, Brand-Gruwel, & Kammerer, 2020). By comparing the learning outcomes of users searching on a smartphone versus a laptop, that search device effect had a non-significant effect on the participant's learning outcome. The device had however a significant effect on the user's behavior ; for example, findings indicated that students used more queries when using their laptops than on smartphones.

The impact of note-taking features in search tools on learning has been examined in several studies. One such study by Freund and Staubach investigated whether participants could add "sticky notes" to articles (Freund et al., 2016) and found that the tool did not have a significant impact on their performance. On the other hand, Roy *et al.* investigated the effect of two active reading tools, highlighting and note-taking, on user learning (Roy, Torre, Gadiraju, Maxwell, & Hauff, 2021). They measured user learning by asking participants to write post-task summary essays and found that neither tool improved vocabulary learning. However, they observed that the highlighting tool allowed participants to cover more subtopics in their essays, while the note-taking tool enabled them to include more facts in their essays.

Câmara *et al.* investigated the efficacy of three interfaces that scaffolded users during complex search tasks (Câmara, Roy, Maxwell, & Hauff, 2021b). The first interface showed users a list of pre-defined relevant subtopics for their *queries*, while the second interface displayed a manually curated static list of relevant subtopics of the searched *topic*. The third interface added feedback on the user's exploration in the topic space. However, these novel features did not significantly improve learning outcomes. Instead, participants were found to explore more subtopics superficially, as evidenced by their increased number of search results viewed but shorter dwell times. This suggests that feedback features can have unintended effects on searchers, leading them to pursue strategies that undermine the depth of their learning.

3.4.3 Searcher cognitive capacities

Individual users of information systems have different levels of cognitive abilities. These abilities affect not only the search performance (Allen, 1998, 2000) and behavior, but also the resulting knowledge acquisition process. These abilities, as per (K.-S. Kim & Allen, 2002), are taken together as "intelligence".

Previous work studied the impact of the users' cognitive abilities on the search behavior and on the ability to find the right information. Perceptual speed, for example, measures the user's ability to quickly and accurately compare similarities and differences among sets of letters, numbers, objects, pictures, or patterns (Ekstrom, French, Harman, & Dermen, 1976).

Working Memory Capacity (WMC) (Baddeley, 2007) is defined as a cognitive processing resource of limited capacity that involves the simultaneous storage and processing of information. It is the small amount of information that can be held in mind and used in the execution of cognitive

tasks. Studies performed by Gwizdka (Gwizdka, 2010, 2017) proved that users with higher working memory have the ability to perform more search actions and achieve their search tasks faster. Sharit *et al.* found a positive correlation between the WMC and the number of Websites visited during a health-related search session (Sharit, Hernández, Czaja, & Pirolli, 2008).

Reading Comprehension Ability (RCA) refers to the capacity to understand written language and extract meaning from it. It involves not only recognizing and decoding individual words and sentences but also comprehending the overall message, purpose, and implications of a text. It was also proved to have a positive relation with search learning outcomes in hypertext environments (Naumann, Richter, Christmann, & Groeben, 2008 ; Coiro, 2011). Studies by Pardi found that working memory capacity and reading comprehension ability were predictive of learning outcomes (Pardi, von Hoyer, Holtz, & Kammerer, 2020).

Furthermore, the user's curiosity can also impact their interest and engagement with information systems, and this can be measured by analyzing Functional Magnetic Resonance Imaging (fMRI) signals (van Lieshout, Vandenbroucke, Müller, Cools, & de Lange, 2018). Additionally, research has found that dyslexic users, compared to non-dyslexic users, tend to look at more documents during each search iteration, despite conducting fewer searches and visiting fewer total documents (MacFarlane et al., 2010). Pupillometry and head distance to the screen have been used to predict the user's skill acquisition during information visualization tasks (Toker, Lallé, & Conati, 2017). These neuro measures offer researchers the ability to capture detailed and nuanced reactions related to the cognitive and emotional activities of the user and could provide a more solid physiological foundation for SAL research. However, the cost of obtaining equipment for these methods is often high, which can hinder researchers from using them.

3.4.4 Knowledge state and familiarity

The user's knowledge state and familiarity with the searched topic play an important role in the learning journey of the user. Research in this area has been actively trying to understand the relation between the user's knowledge state, search behavior, and learning progress. Previous research proved the impact of the user's knowledge on search behavior (Xie & Joo, 2012 ; K.-S. Kim & Allen, 2002), and could identify two types of user knowledge : domain knowledge and task topic knowledge (Li & Belkin, 2008). Domain knowledge is one's knowledge of a general subject domain of the search task, while task topic knowledge is one's knowledge of the specific search task topic. Users familiar with a specific task find it less difficult when performing it (Byström & Järvelin, 1995). The interaction of these two types of knowledge supports the user's behavior when they are achieving a search task (Marchionini, 1993).

It has also been found that users with higher domain knowledge submitted more queries (X. Zhang, Anghelescu, & Yuan, 2005 ; Monchaux, Amadieu, Chevalier, & Mariné, 2015 ; Sanchiz, Chin, et al., 2017) and longer ones (White, Dumais, & Teevan, 2009 ; Tamine & Chouquet, 2017 ; Kang & Fu, 2010) than non-experts did. They also used wider (Monchaux et al., 2015 ; Sanchiz, Chevalier, & Amadieu, 2017), more specific vocabulary (Vakkari, Pennanen, & Serola, 2003 ; O'Brien, Kampen, Cole, & Brennan, 2020a), more effective (Sihvonen & Vakkari, 2004) and less combination (Hsieh-Yee, 1993) query terms than non-experts. As for the time spent on Webpages, experts spent less time (Duggan & Payne, 2008) visiting a page. When Zhang *et al.* investigated the origin of the query terms (Y. Zhang & Liu, 2020), they found that more than half of them were from the prior knowledge of users, as represented in their mind map.

Also, previous studies have found that users with high topic knowledge tend to have longer and more complex queries (Hembrooke, Granka, Gay, & Liddy, 2005), use more search expressions (Allen, 1991), spend less time reading documents, and save more documents (Kelly & Cool, 2002). Roy *et al.* investigated at which time during a search session learning occurs, and found that the learning curve is largely influenced by a user's prior knowledge of the searched topic (Roy, Moraes, & Hauff, 2020).

Users with different knowledge gain can have different search behavior. A recent study (Bhattacharya & Gwizdka, 2019; Gwizdka & Chen, 2016) found that the reading behaviors of high knowledge gain users and low knowledge gain users differ significantly. The results showed that participants with a higher change in verbal knowledge differ by reading significantly less and entering more sophisticated queries, compared to those with a lower change in knowledge. Collins-Thompson *et al.* studied the impact of distinct query types on knowledge gain (Collins-Thompson et al., 2016). They investigated potential indicators of learning in Web search, effective query strategies for learning, and the relationship between search behavior and learning outcomes through a lab-based user study.

3.5 The cost of finding information

Searching for information on a search engine can entail certain costs. These costs are not monetary but rather in terms of time and effort. They can pose a significant challenge, particularly when using search engines as a learning tool. This section will examine the various costs of accessing information on a search engine for learning, such as reformulating different queries, reading different numbers of pages, sorting through redundant information in various pages, spending more time than necessary, and losing motivation due to the difficulty of accessing information.

3.5.1 Reformulating different queries

When searching for information on a search engine, users often face the challenge of reformulating different queries. This is because the search engine may not always provide accurate or relevant results for the initial query. Guan and Cutrell reported that when users could not find the needed information in search results, they either selected the first result or switched to a new query (Cutrell & Guan, 2007). Reformulating the query multiple times can be time-consuming and frustrating, especially when trying to learn something new. The findings presented in (Bailey et al., 2012) revealed that users tend to issue the highest number of queries when attempting to learn how to perform a task, with an average of 13 queries. Similarly, a considerable number of users issue an average of 6.8 queries when trying to discover more information about a specific topic.

3.5.2 Reading different numbers of pages

Another cost of accessing information on a search engine is the need to read different numbers of pages. When searching for information on a search engine, users often have to navigate through multiple pages of search results to find the relevant information. This can be a challenge, especially when dealing with a large amount of information. For example, imagine a person searching to learn some information on a medical condition. The search engine may provide thousands of results, and the user may have to navigate through multiple pages of search results to find the relevant

information. This can be time-consuming and frustrating, and it can make learning a challenging task.

3.5.3 Sorting through redundant information in various pages

Crawling the enormous amount of data on the Web can be an intimidating task, especially when crawling makes up approximately 96% of all Web content (Madhusudan & Poonam, 2017). In addition to navigating through multiple pages of search results, users also have to sort through redundant information in various pages. This is because search engines often provide multiple results for the same query, and some of these results may contain redundant information.

3.5.4 Spending more time than necessary

Another cost of accessing information on a search engine is the need to spend more time than necessary. This is because search engines may not always provide accurate or relevant results for the initial query, which may result in users having to spend more time searching for relevant information. This can be a challenge, especially when dealing with a large amount of information. According to (Bailey et al., 2012), users spend an average of 13.5 minutes when trying to achieve educational tasks and learn about a specific topic. This emphasizes the importance of taking advantage of the limited effort and time that users are willing to spend on a tool to find information before they give up.

3.5.5 Losing motivation due to the difficulty of accessing information

Finally, one of the most significant costs of accessing information on a search engine for learning is the potential to lose motivation due to the difficulty of accessing information. This is because searching for information on a search engine can be challenging and time-consuming, and it can make learning a challenging task. When learners face these challenges repeatedly, they may lose motivation and turn to other sources for help. Griffiths and Brophy's study (Griffiths & Brophy, 2005) investigated the reasons why academic students abandon information search. The results showed that 30% of the participants were unsuccessful in locating the required information. Additionally, 12% of the users gave up their search because they experienced frustration and felt that they had searched everywhere possible or did not know where else to look. These findings suggest that a significant number of searchers face difficulties in finding the information they need using conventional Web search engines.

Through an analysis of eye tracking behavior during search sessions, Jiang and colleagues discovered that users engaging in goal-oriented tasks exhibited a reduction in the number of eye fixations as the search session advanced (Jiang, He, & Allan, 2014). This could suggest a loss of interest, an increase in mental effort, and possibly even fatigue or boredom.

The importance of understanding why searchers end their sessions can vary between having satisfied their need, lack of time, or giving up. This has been an area of investigation to identify typical abandonment points during the information search process (Maxwell & Azzopardi, 2018), and to automatically determine indicators to identify good or bad abandonment for different search contexts (K. Williams & Zitouni, 2017).

3.6 Modeling and predicting user knowledge

In Section 3.4, we discussed the impact of knowledge on the user's learning behavior and how the user's behavior can indicate their knowledge state or knowledge gain. In this section, we will focus on two key concepts that are essential for adapting and personalizing search results : knowledge modeling and knowledge gain prediction. The first subsection will explore knowledge modeling, which involves creating a structured representation of the user's existing knowledge. The second subsection will review previous work on predicting knowledge state representation and knowledge gain, which involves forecasting the amount of knowledge the user will acquire by engaging with a particular search query or learning activity.

3.6.1 User knowledge modeling

User profiling of interest discussed in Section 2.2.3.2 and knowledge representation are two concepts that share some similarities in the context of personalized search. User profiling of interest involves creating a profile based on the user's past behavior, such as search history, clicks, and dwell time. On the other hand, knowledge representation focuses on creating a structured representation of the user's existing knowledge. This specific task of knowledge representation has not received much attention from the search as learning community yet. While knowledge representation has been a subtask of other prediction tasks, it has not been extensively explored as a standalone area of research. Two commonly used methods for modeling the user's knowledge are Knowledge Tracing (KT) and Item Response Theory (IRT). These methods mainly differ in the type of data they use and the assumptions they make about how knowledge is acquired and assessed.

Knowledge tracing is the process of representing the knowledge needed to master a domain and diagnosing user knowledge states through online user behavior traces, creating a profile of those knowledge elements. Sluis and Broek proposed a method for representing a user's knowledge based on the queries they submit, assuming that a user's search history in IR reflects their knowledge (Sluis & Broek, 2010). They used synsets, which are sets of synonyms representing a single concept in a semantic network, to represent the user's knowledge. For each query submitted by the user, the set of synsets touched upon was computed as the union of the synsets related to each word in the query.

$$Sq(q) = \bigcup_{w \in q} S(w) \tag{3.1}$$

where $S(w)$ gives the set of all possible synsets s of the lexical dictionary W (i.e., WordNet) related to the word w of query q. To estimate the user's knowledge on a particular topic, the semantic distance between the topic and the user's knowledge model was calculated as a weighted count of the related synsets. A decay function was applied to account for the fact that a user is unlikely to have as much knowledge about synsets that are several steps away. The function assigns a high penalty to synsets that are far away from the user's knowledge. The resulting value ranges from a minimum of 0 to a maximum of 2.00.

Bayesian Knowledge Tracing (BKT) (Corbett & Anderson, 1994) is a statistical algorithm used based on Bayesian inference, which allows for the probability of a student knowing a particular concept to be updated based on their performance on subsequent questions or tasks. BKT takes into account both the correctness of the student's response and the likelihood that they guessed the correct answer. By continually updating the student's knowledge estimates, BKT can make

more accurate predictions about their future performance and adapt the learning experience accordingly.

Item Response Theory IRT is a mathematical model that analyzes the relationship between an individual's latent ability and their performance on test items or questions. The basic idea is to calculate the probability of a user answering a question item correctly (Kline, 2005). Each "item", also known as a question, is characterized by a set of parameters, including its difficulty level and discrimination power. The difficulty level of an item is defined as the level of the underlying ability or trait required to have a 50% chance of answering the item correctly. A more difficult item requires a higher level of ability or trait to answer correctly. The traditional IRT model mentions a single θ characteristic, and is represented as follows :

$$P(Y_i = 1 \mid \theta_i, \beta_i) = \frac{1}{1 + \exp(-[\theta_i + \beta_i])} \tag{3.2}$$

In this equation, θ is the user's latent knowledge ability, and $P(Y_i = 1 \mid \theta_i, \beta_i)$ represents the probability of individual i answering a particular question correctly given their ability θ_i and the difficulty of the question β_i. The equation uses the logistic function, represented by the sigmoid-shaped curve defined by the exponential function $e^{-[\theta_i+\beta_i]}$. This function maps the difference between the individual's ability and the question difficulty onto a probability scale that ranges from 0 to 1. The denominator of the equation, which is equal to $1 + \exp(-[\theta_i + \beta_i])$, normalizes the probability so that it falls within this range.

Syed and Collins-Thompson utilized the IRT as the foundation of their model and presented a function that takes into consideration various parameters (Syed & Collins-Thompson, 2017b). These parameters included the user's individual learning rate, a weight assigned to the term in the *expert* model, the item parameter for the tested item, the ease of learning of the term, and the frequency of appearance of a term. To determine the probability of a correct response, the function utilized a linear combination of these parameters and created an algorithm that maximizes the learning and minimizes the effort spent.

3.6.2 Predicting knowledge state and knowledge gain

The user's *knowledge state* refers to their level of understanding or what they already know. On the other hand, *knowledge gain* represents the increase in the user's knowledge state resulting from their interaction with a search tool. In the context of this thesis, we refer to this process as *learning*, and therefore, we will use the *knowledge gain* and *learning outcomes* interchangeably. We will discuss two approaches for predicting a user's knowledge. The first and most common approach is to classify according to their knowledge level (ie. expert or novice), which has been widely used in various domains, such as medicine, law, and information retrieval. The second approach, although less common, aims to quantify the user's knowledge of a specific topic.

Classifying user knowledge. White et al. involved a large-scale analysis of real-world search data interactions (White et al., 2009), with over 500,000 unique users visiting more than 10 billion URL pages. Using a binary classifier with search log data, the researchers were able to predict users' domain expertise. The users' knowledge levels were measured using a binary method that categorized them as novices or experts, based on their visits to specific Websites that were judged advanced by experts in the experiment's domains : medicine, finance, legal, and computer science. Another work by Liu *et al.* proposed several prediction models that use logistic regression to distinguish between novice or knowledgeable users (J. Liu, Liu, & Belkin, 2016b). The latter

models rely on analyzing the user's behavior during three different stages of their search process : the first round of queries, the middle of the session, and the end of the session. The study's findings suggest that early-session behavioral variables, such as the length of the user's initial search query, the duration of their first visit to the search engine results page, and the duration of their first visit to the first document, can reasonably predict the user's knowledge level. However, behaviors during the later stages of the session had lower prediction performance.

Outside the search context, expert-finding methods in the literature involve identifying experts based on text or documents in various domains such as law, medicine, and more. These methods rank potential experts based on the probability of their expertise in a given topic, which reflects the probability $p(ca|q)$ of a candidate ca being knowledgeable in a topic q. To address this, Balog *et al.* proposed an approach that utilizes generative probabilistic modeling with candidate and document, represented in language models, to identify suitable experts based on users' queries (Balog, Azzopardi, & de Rijke, 2009).

Quantifying user knowledge The user's domain's knowledge (DK) was measured as a continuous number by Zhang *et al.* who investigated the use of implicit behavioral features, such as the average query length and the rank of documents, to predict the user's domain knowledge (X. Zhang, Cole, & Belkin, 2011 ; X. Zhang, Liu, Cole, & Belkin, 2015). The study was conducted using recall-oriented search tasks in the genomic domain. They employed a multiple regression model that utilized the average query length and the average rank of documents consumed in the search engine results page (SERP). The best model identified three behavior variables as predictors of domain knowledge : the number of documents saved (Nb_{Saved}), the average query length (Avg_{qlen}), and the average ranking position of documents opened in the SERP. The estimated domain knowledge numbers were validated against users self-rating to their expertise and familiarity with the search topic. The model is expressed below :

$$DK_{Zhang} = -1.466 + 0.039 * Nb_{Saved} + 0.147 * Avg_{qlen} + 0.130 * Avg_{rank} \tag{3.3}$$

Term-based methods are based on statistical metrics that track the frequency of the terms in the *visited documents.* Bharat and Mihaila computed the score of an expert as a 3-tuple of the form (S_0, S_1, S_2) (Bharat & Mihaila, 2001), where S_i component considers only key phrases containing the query terms k_i. For example, S_0 is the score computed from phrases containing all the query terms.

$$S_i = \sum LevelScore(p) * FullnessFactor(p, q) \tag{3.4}$$

$LevelScore(p)$ is a score assigned to the phrase by virtue of the type of phrase it is (title, headings, anchor text..) and $FullnessFactor(p, qh)$ is a measure of the number of terms covered by the terms in q. In an example where the number of keywords within any key phrase was limited to 32 and the used $LevelScore$ was set to 16 for title phrases, 6 for headings, and 1 for anchor text the score of the expert was defined as follows :

$$Expert\_Score = 2^{32}S_0 + 2^{16}S_1 + S_2 \tag{3.5}$$

Yu *et al.* considered features extracted from user interactions with search results and resources, such as query reformulations, clicks on search results, and time spent on Web pages, to predict the user's knowledge state (R. Yu, Gadiraju, Holtz, et al., 2018b). The features were selected based on their inter-correlation and their correlation to the prediction goal. The study employed a supervised machine learning approach, utilizing a classifier to predict the user's knowledge state,

which was classified into three classes : low, medium, and high. The authors trained and evaluated several models to predict the user's knowledge state on three different dimensions : pre-knowledge state, post-knowledge gain, and overall knowledge state. The real user's knowledge was measured using true-false questions, and the knowledge gain was determined by the difference between the post- and pre-scores. The authors proposed a topic-independent modeling approach for predicting the user's knowledge state in learning-oriented search sessions. The authors have presented a closely related work (R. Yu, Tang, Rokicki, Gadiraju, & Dietze, 2021a) where they have used the same dataset as in their previous work and combined the 70 user behavior features with 109 Web resource-centric features. These new features included aspects such as the text-linguistic tone and its complexity, as well as the structural aspect of the HTML pages. Feature selection methods were used to reduce the number of features, and the knowledge gain was predicted using machine learning classifiers such as Bayes, Logistic Regression, Support Vector Machine, and Random Forest. Their results show that this approach outperformed their previous work.

3.6.3 Comparing knowledge modeling and interest modeling

While a significant amount of research has concentrated on modeling a user's interests in information retrieval, it is becoming increasingly important to model and personalize search results based on the user's knowledge. In the following, we share our thoughts about several key differences between modeling knowledge and interests.

The dynamic nature of knowledge One of the key differences between knowledge and interests in information retrieval is their dynamicity. Knowledge is constantly changing and evolving as the user reads and learns new information, whereas interests are generally more stable over time. Brookes formulated their "fundamental equation" of information and knowledge, stating that exposure to information changes an information searcher's current state of knowledge to a new knowledge structure (Brookes, 1980). Marchionnini described information seeking as "a process" in which humans purposefully engage in order to change their state of knowledge (Marchionini, 1995). This dynamism has important implications for the design of information retrieval systems and the modeling of user behavior. This means that the knowledge that a user has today may be different from what they had yesterday or a week ago. In contrast, interests are generally more stable over time. It takes time for a user to develop an interest in a particular topic, and this interest may persist for years.

The granular nature of knowledge modeling User knowledge is granular because it consists of specific facts and concepts that the user has learned over time. To provide novel, non redundant search results that will help the user learn, it is necessary to model this knowledge in a granular structure to capture the specific details of what the user knows and what they don't. On the other hand, to represent the interest, a high-level model may be enough by identifying the topics or categories that a user is interested in, such as sports, music, or politics. This can be done by analyzing the topics of the Web pages that the user visits, the queries they submit, or other user behavior data.

The role of sources Another important difference between knowledge and interests in information retrieval is the sources that are used to model them. In search as learning, the sources for knowledge are limited and usually derived from the documents or Web pages that the user visited.

When it comes to modeling user interests, research has explored a wider range of sources to infer this interest like the content of the visited Web pages (Matthijs & Radlinski, 2011), and other user behavior data such as search queries (Bilenko et al., 2008), clicks, social media activity, and even eye-tracking data (Bhattacharya & Gwizdka, 2019).

Objective behind the search sessions In search as learning, users typically have a specific information need or learning goal in mind when they conduct a search, which makes their session oriented to achieve this learning task. Therefore, the judgment of a relevance of a document should not only account for the user's knowledge when they start the session, but also the knowledge state they would like to achieve of doing the search. In contrast, user interests are often not as clearly defined or goal-oriented as they may be based on personal preferences, hobbies, or curiosity, and may not be related to a specific learning goal or information need.

3.7 Assessing learning through experimental measures

Measuring the user's knowledge gain or learning outcome is a common step in SAL experiments. This measurement is usually achieved by comparing the user's state before and after the search session. We present in this section the various methods that have been used to measure learning during search.

3.7.1 When to measure

Before and after the session One common approach is to measure a user's knowledge before and after a search session using scores or quantified measures, as discussed in the previous section. This approach provides a snapshot of the knowledge states at two points in time. By comparing these scores or snapshots, knowledge gain can be assessed and quantified. For example, Yu *et al.* measured the user's knowledge using multiple choice questions before and after the search session and calculated the knowledge gain as the difference between the scores of the assessments (R. Yu, Gadiraju, & Dietze, 2018). Câmara *et al.* also measured the user's vocabulary familiarity with topic-related terms at two points in time to assess the learning outcomes resulting from using their proposed interface (Câmara et al., 2021a). Bhattacharya and Gwizdka asked their study participants to write summaries about their knowledge about a topic before and after the search session, and then measured the vocabulary learning using word embeddings (Bhattacharya & Gwizdka, 2019).

During the session. Another method is to measure the user's knowledge *during* the search session to closely understand how knowledge is acquired during the session. We mention a limited number of studies that have utilized this approach. Roy *et al.* interrupted users at five-minute intervals during their search sessions to assess their vocabulary learning (Roy et al., 2020). Liu *et al.* asked users to draw a mind map during the session and update it as they learned new information (H. Liu, Liu, & Belkin, 2019). These methods allow researchers to capture the learning process and determine at what point the user is gaining new knowledge. While these methods provide a closer understanding of the evolution of the user's knowledge, the fact that they interrupt the user during their session risks missing out on capturing the natural search behavior.

Deferred Measurement. Deferred measurement is a way to assess the long-term retention of knowledge over time. This method involves measuring a user's knowledge at a later point in time,

such as a week or a month after the search session. Wildemuth (Wildemuth, 2004) conducted a study to measure the long-term knowledge retention of 77 medical students on three separate occasions : before they entered a medical course, immediately after the course ended, and six months after the completion of the course. Similarly, Qiu *et al.* (S. Qiu, Gadiraju, & Bozzon, 2020) investigated the retained knowledge gain of participants in a knowledge test. To measure retention, the authors considered the number of questions that were correctly answered in the immediate post-test but were answered incorrectly in a long-term memory test. In the study conducted by Syed and Collins-Thompson (Syed & Collins-Thompson, 2018), the participants were given a delayed post-test nine months after the experiment to analyze how much of their initial vocabulary knowledge they retained over time and to investigate the effectiveness of their personalized retrieval models on long-term retention.

3.7.2 How to measure

A wide range of methods were used in SAL experiments to measure the user's knowledge gain. We follow the classification proposed by Urgo and Arguello, which encompasses the nine different categories of assessment (Urgo & Arguello, 2022) : (1) self-report, (2) implicit measure, (3) multiple-choice, (4) short-answer, (5) free recall, (6) sentence generation, (7) mind map, (8) argumentative essay, and (9) summary and open-ended. We briefly explain in the next paragraphs each of them.

Self-reported measures typically ask study participants to rate their own learning using a Likert scale, and this measure is characterized by its ease of development. Participants can be asked to report their learning performance (Collins-Thompson et al., 2016), topic familiarity (Ghosh, Rath, & Shah, 2018 ; J. Liu, Belkin, Zhang, & Yuan, 2013), prior and/or post-knowledge (Mao et al., 2017), or knowledge gain (Ghosh et al., 2018). Another way to explicitly measure knowledge state is by asking users about their familiarity with thesaurus terms in a domain (X. Zhang et al., 2005 ; Câmara et al., 2021a ; Cole, Gwizdka, Liu, Belkin, & Zhang, 2013). The thesaurus term familiarity method provides a relatively objective way to test users' knowledge because the terms stand for distinct concepts in the domain. However, one main drawback of these methods is that reported scores can be influenced by individual factors such as the gender of the participant (González-Betancor, Bolívar-Cruz, & Verano-Tacoronte, 2019) and the level of anxiety (Colbert-Getz, Fleishman, Jung, & Shilkofski, 2013).

Although subjective assessments, like self-reported measures, have been criticized (X. Zhang et al., 2015) and may be prone to inaccuracies, research from educational psychology (Mitrovic & Martin, 2007 ; Malabonga, Kenyon, & Carpenter, 2005) and information science (Kelly, Kantor, Morse, Scholtz, & Sun, 2006 ; X. Zhang et al., 2005) has shown that self-assessment can lead to consistent results in assigning knowledge levels, particularly in specific domains. For example, studies have found a correlation between self-assessments of task knowledge and knowledge of medical and biology thesauri in specific domains (Cole et al., 2013).

Implicit measures capture the behavior of participants during search sessions to estimate their knowledge state, such as query and click complexities (Chi, Han, He, & Meng, 2016). The measurements resulting from these methods are not influenced by individual factors since implicit measures do not ask users to report their knowledge state or gain. Additionally, the estimations are generated by the same system and measurement is normalized, and they allow for timely feedback with measurements captured on the fly. However, these methods are not commonly used

yet, possibly because they are not valid alone and need to be coupled with other measurements (Chi et al., 2016).

Multiple-choice assessments ask participants a set of close-ended questions proposing correct and incorrect options, with predefined correct answers. Another form is the True-False-I don't know options (R. Yu, Tang, Rokicki, Gadiraju, & Dietze, 2021b; Gadiraju et al., 2018; Xu, Zhou, & Gadiraju, 2020; S. Qiu et al., 2020; Kalyani & Gadiraju, 2019), where the knowledge gain is calculated as the difference between the post and pre-scores. Other studies included multiple-choice answers in their tests. The knowledge gain was measured either as the difference between scores (von Hoyer, Pardi, Kammerer, & Holtz, 2019) or the number of questions answered incorrectly (Syed & Collins-Thompson, 2017b).

Short-answer assessments involve asking questions that are open-ended but have a relatively short and objectively correct answer (Duggan & Payne, 2008). Short-answer questions do not ask participants to make one or more selections from a predefined set of options. Instead, participants are required to generate a response completely on their own. It can be useful to test the user's knowledge on factual information, and assessment scores can be easily compared across participants (Davies, Butcher, & Stevens, 2013).

Free recall assessments involve asking participants to list as many important terms, phrases, or facts related to the topic of a search task. Knowledge gains can be measured in different ways. For example, Bhattacharya *et al.* (Bhattacharya & Gwizdka, 2019) asked their study participants to free-recall as many words or phrases on a pre-defined topic as they could. The knowledge change was then estimated as the semantic similarity between the text provided and a vocabulary of expert words. They also used the change in the number of words as well as the angular similarity between the text provided by the users about their knowledge and a vocabulary of expert words.

Sentence generation assessments require individuals to produce short sentences or definitions to evaluate their vocabulary knowledge of specific terms. The sentences should be grammatically correct and demonstrate their understanding of the term. These tests are easy to create and can prevent guessing from a predefined set of answers. However, comparing participants' responses can be challenging. Additionally, they cannot measure the user's higher cognitive abilities when using the learned terms.

Mind map are a visual representation of information that revolves around a central concept. In the context of search as learning, mind maps have been utilized to comprehend the learning process during search sessions, more than actually quantifying it. For instance, Liu and colleagues (H. Liu et al., 2019) asked participants to draw a mind map based on their pre-existing knowledge of a particular topic and then modify it during their search session. The study analyzed the changes in the type of changes on the graph, such as adding, modifying, and deleting nodes, at various stages of the search task. Similarly, Zhang and Liu (Y. Zhang & Liu, 2020) also employed mind maps to understand the participants' prior knowledge of the subject by requesting them to create an initial map. They considered that the user's previous knowledge is the set of terms in their initial map before the search session. They then compared the bag of words of the map nodes to the query terms to establish the relationship between the previous knowledge and the search behavior. Other experimental studies (Saito et al., 2011), compared the concept maps drawn by users before and after their search sessions. They identified the number of concept words (nodes), links between nodes, and link words to understand the structural changes in the user's knowledge during the search sessions. While mind maps can effectively illustrate how users organize their knowledge, they may not be appropriate for experiments involving participants who lack familiarity with the domain or mind maps. Furthermore, comparing two mind maps can present difficulties.

Argumentative essays require participants to write an essay presenting arguments both for and against a position of the topic. Essays can be graded based on the number of valid pro and con arguments provided (Demaree et al., 2020), and even though answers are not pre-defined, it is still possible to compare and grade them fairly across participants. Additionally, both argumentative essays and short-answer assessments are considered close-ended, which minimizes guessing as they do not provide a pool of options to choose from.

Open-ended assessments and summaries require participants to either summarize their knowledge on a particular topic or provide responses to open-ended questions (Abualsaud, 2017). Unlike short-answer questions, there is no specific expected answer. This type of assessment measures a higher level of learning and often asks users to write about what they have learned during their search. For example, Kalyani and Gadiraju (Kalyani & Gadiraju, 2019) asked users to design a plan that they manually checked and tagged. Measuring learning from open-ended responses can be more challenging. To address this, studies have used a variety of grading strategies. Wilson and Wilson (Wilson & Wilson, 2013) proposed a qualitative coding scheme to evaluate open-ended written summaries in which participants describe what they have learned. These measures indicate the depth of learning shown in the three levels of learning in Bloom's taxonomy : understanding, analysis, respectively described as follows : The first measure assesses the quality of facts recalled in the summary, the second measure the interpretations in the recalled facts and the third identified statement that compared facts or used facts to raise questions about other statements.

Although many studies have used only one method to measure user knowledge, some studies have combined multiple techniques to capture different aspects of knowledge. For instance, some studies have used close-ended assessments to capture simple cognitive knowledge, while open-ended assessments were used to capture complex knowledge (Urgo & Arguello, 2022). Additionally, summary and open-ended assessments have been commonly used together with short-answer assessments, and with self-report assessments to compare actual learning to perceived learning.

3.8 Detecting and adapting learning-oriented sessions

As discussed in Section 3.2.2, several Web search taxonomies have been proposed. They classify the purposes for which users employ search tools, and they generally agree on three broad categories of search sessions : transactional, navigational, and informational. Understanding the user's search needs and objectives, as well as the specificity and complexity of their search task, is crucial in the development of effective search systems. Ingwersen emphasized the importance of understanding the user's tasks or needs behind their interaction with the system for productive information retrieval (Ingwersen, 1992). Saracevic and Kantor demonstrated that the specificity and complexity of the search task have a significant impact on search performance (Saracevic & Kantor, 1988).

In order to adapt the search session to be oriented toward learning and to apply the necessary algorithms, it is important to identify the session as a learning or goal-oriented objective one. Within the Broder taxonomy, informational search sessions are those that involve learning intentions, where users seek to acquire new knowledge about a particular topic.

To identify learning-oriented sessions, two methods are used in combination : session segmentation and intent detection. Session segmentation (Hagen, Gomoll, Beyer, & Stein, 2013) refers to the process of dividing a user's search session into segments based on certain criteria, such as time

intervals or changes in the user's query patterns. This segmentation can be useful in identifying different types of search behavior within a single session, such as when a user switches from an informational search to a transactional search. Intent detection, on the other hand, involves identifying the user's underlying goal or intent behind their search queries. A common method in intent detection is automated classification approaches such as supervised (Jansen, Booth, & Spink, 2008; Kravi et al., 2016) and unsupervised (Baeza-Yates, Calderón-Benavides, & González-Caro, 2006; Kathuria, Jansen, Hafernik, & Spink, 2010) approaches, which have been applied to classify Web search queries. The features used in the classification approaches are extracted from query terms (Kravi et al., 2016; Hu, Wang, Lochovsky, Sun, & Chen, 2009), user-click behaviors (Kravi et al., 2016; Y. Liu, Zhang, Ru, & Ma, 2006), anchor links (Y. Liu et al., 2006; Lee, Liu, & Cho, 2005), Web documents' content (Jansen et al., 2008), and page views (Kravi et al., 2016; Kathuria et al., 2010). For example, Yu *et al.* used supervised models for classification, where they extracted 22 features based on multiple dimensions of a search session (R. Yu, Gadiraju, & Dietze, 2018). These features were structured into three categories : features related to query (i.e., features related to the number of query terms and the between query similarity), session (i.e., total number of queries issued, session duration related and session breaks related features), and browsing behavior (i.e., features related to the number of clicks, revisited pages, and similarity between the query and the clicked URL). These features were used in some supervised models to classify the session intent. The results showed reasonable results in detecting informational goals, and some ambiguity in detecting transactional sessions. For example, the intention behind some query like *"best universities for computer science"* can be confusing between informational and navigational : The user may be looking for information about the top universities that offer computer science programs, but may also be looking to navigate to the Websites of those universities to get more detailed information about their programs, faculty, admissions requirements, etc.

Some existing work has proposed adapting search tools to help users achieve their search goals. For example, Urgo and Arguello developed a *Subgoal Manager* that breaks a learning-oriented search task into smaller subgoals (Urgo & Arguello, 2023). This tool allows users to develop subgoals, take notes related to specific subgoals, and mark subgoals as complete. Searchers can edit, delete, and add new subgoals throughout the search session. In addition, Câmara *et al.* proposed a search design that scaffolds users during their search for complex tasks by presenting potential subtasks that need to be achieved (Câmara et al., 2021a).

Contributions

CHAPTER 4

Study I : Exploring Different Representations for User Knowledge and Needs

Learning involves the transition of a user's knowledge from one state to another. In order to support the user's learning during the search sessions, it is essential to have a granular representation of their knowledge and needs. In the field of search as learning, this aspect has typically been examined only as a part of other algorithmic or optimization processes, rather than being studied in isolation and compared with different methods. Nonetheless, it has not been given significant attention as a standalone subject of study. This chapter proposes a framework, `RULK`*, for representing the user's knowledge during the search sessions. The framework maintains an internal representation of the user's knowledge state, which is continually updated as the user progresses in their search.* `RULK` *tracks the user's knowledge level regarding a need of a particular topic. The chapter implements three variations of* `RULK` *: one based on keywords, another using large language models, and the third using named entities. The framework's effectiveness is evaluated by estimating the user's knowledge gain, also refered to as the learning outcome, and comparing it with the real user knowledge gain. The correlations between the estimated and knowledge gains are also analyzed.*

4.1 Introduction

The field of search as learning (SAL) involves understanding the process of human learning during search sessions and adapting tools to better support the related knowledge acquisition process. Research in this area has primarily focused on analyzing the factors that affect learning and identifying the behavioral characteristics that can predict a user's level of knowledge. One common area of study is the analysis of user behavior during search sessions, including factors such as query length (X. Zhang et al., 2011, 2015; Balog et al., 2009), duration of visited pages (J. Liu et al., 2016b), and visits to specific websites (White et al., 2009), which can be used to distinguish between novice and expert users. Other studies have employed machine learning models to estimate the learning outcome at the end of a session (R. Yu, Gadiraju, Holtz, et al., 2018b; Otto et al., 2021). These studies have shown promising results in defining the knowledge level of a user at the end of a session.

The process of learning during search is characterized by a transition from one knowledge state to another. The mentioned knowledge change during search does not occur suddenly between the start and the end of the session. Instead, it is a gradual process, and every document that is read may contribute to this change. We argue that it is important to track the user's knowledge and trail any changes that occur over time, especially given that this knowledge constantly evolves with exposure to new information. The majority of previous work, however, typically focused only on predicting the user's knowledge level with respect to a concept, rather than tracking what they know or don't know. By tracking the user's knowledge state throughout the search session, we can identify what the user knows and what specific knowledge is needed to fill those knowledge gaps. This information can also help in selecting which documents to present to the user, and what specific parts of the document are relevant to their learning objectives. Thus, tracking changes in the user's knowledge state can help in providing more effective and personalized support for the learning process during search. Additionally, the majority of the studies often measure the user's knowledge with respect to a general topic or concept, rather than a specific set of information to learn.

The search as learning community recognizes that users have their own internal model of the world, but there has been limited effort in developing a suitable approach for capturing and representing this knowledge in research. Additionally, the representation structure has often been treated as a secondary task and has not received adequate attention on its own. Syed *et al.* (Syed & Collins-Thompson, 2018) employed probabilistic methods from the field of Intelligent Tutoring Systems (ITS) in education to predict the probability of a user answering a given question or item correctly. Sluis et al. used synsets extracted from user queries to model their knowledge (Sluis & Broek, 2010). The research community gave little attention to the development of a comprehensive approach for capturing and representing users' knowledge. Considering the limited amount of existing research on this topic, and the significance of exploring diverse methods for representing users' knowledge during search sessions, we emphasize the need for understandable representations.

To address this gap, we propose a novel framework for **R**epresenting **U**ser **L**earning and **K**nowledge (`RULK`). This framework uses understandable representations of the user's knowledge, tracks it, and updates it with the arrival of new information. The framework has the ability to measure the knowledge gain at any point during a search session, with respect to a specific learning goal defined by a set of information pieces. The components of `RULK` are inspired by the steps

taken in modeling user interest and personalization. It involves gathering information about the user, creating representative profiles based on this information, and finally exploiting this profile.

To demonstrate how one could implement RULK in SAL systems, we implement three variations of it : RULK_{KW}, using keyword representations ; RULK_{LM}, using embeddings generated by a large language model ; and RULK_{NE}, using named entities. We examine the behavior of each implementation, particularly in estimating user knowledge, by analyzing logs of real-world user interactions with a search system focused on learning, along with their associated learning scores.

As search sessions extend in duration, users tend to engage with more documents and have more interactions, resulting in heavier knowledge representations that RULK needs to process and represent. In the context of profile-document matching, a previous study (Zamani & Shakery, 2018) found that the effectiveness of matching profiles with documents underwent significant changes as the number of unique words used to profile users increased. In our framework, we also aimed to investigate how the accuracy of RULK is influenced by the length of search sessions and heavier user profiles. This exploration was conducted to gain insights into the framework's capability to effectively to handle large amounts of data. We evaluate this on three aspects : the number of submitted queries, the number of visited documents, and the duration of the session.

We propose in this chapter the RULK framework and its different implementation and user study to evaluate it. We aim to answer the following research questions :

RQ4-1 How can different implementations of RULK accurately estimate the knowledge gain of users during a search session, and how do they compare with each other in terms of their accuracy ?

RQ4-2 Is there an improvement in estimating knowledge gain through combining representations when compared to a single representation ?

RQ4-3 How does the accuracy of the representations used in RULK change with varying session lengths ?

4.2 Notations

d	a document read by the user
$\overrightarrow{v_d}$	A representation of the document d
$\overrightarrow{c_{ks}}$	**c**urrent **k**nowledge **s**tate
$\overrightarrow{t_{ks}}$	**t**arget **k**nowledge **s**tate
RULK	The **R**epresenting **U**ser **L**earning and **K**nowledge framework
RULK_{KW}	An implementation of RULK using **k**eywords
RULK_{LM}	An implementation of RULK using **l**anguage **m**odels
RULK_{NE}	An implementation of RULK using **n**amed **e**ntities

4.3 RULK framework

We present in this section the RULK framework for **R**epresenting **U**ser **L**earning and **K**nowledge. Our framework employs different representations to track the user's knowledge throughout search sessions and dynamically updates this representation as new information becomes available. At any point during the search session, our framework can estimate the user's learning gain in relation to a complex learning topic.

We start by outlining the essential elements required for constructing this framework and for defining the learning goals. We elaborate then on the various components of the framework, illustrating how they dynamically interact with each other.

4.3.1 RULK elements

4.3.1.1 Document as a knowledge source

During the search sessions, users often rely on their prior knowledge to formulate queries. However, the content of the pages they visit and read is the primary source of knowledge for users during their search sessions. The content of the document the user's visit has been little used in user interest profiling in IR for (Matthijs & Radlinski, 2011; Biancalana et al., 2008; Bilenko & White, 2008), we will use it in our work to profile the user's knowledge.

As users navigate through multiple pages, the search results provide new information to the users, which is integrated into their existing knowledge to form an updated state of it. Their knowledge state continues to evolve as they encounter new information. These search results can take the form of snippets displayed on the search engine results page (SERP), as well as the actual content of the page. These pages can be any content the user interacts with during their search session (e.g., Web pages, textbooks, videos, or courses). Without loss of generality, here we focus on the textual content of the pages read by the user, referring to it as a *document* d.

4.3.1.2 Document as a learning objective

In the context of learning, one of the main differences between profiling the user's interests and their knowledge is that creating a user's knowledge profile involves a goal-oriented approach, while interests do not have limits. Interests can vary widely and may not be directly related to the learning objective, while knowledge profiles are tailored to the specific learning objectives of the user. This means that learning should be measured in reference to a specific target. To achieve this, a reference document is needed. It must contain the set information required to form a complete knowledge state. This reference document can be an actual document or a virtual one containing the set of needed information. We refer to this document as the **"target document"** that we consider essential to measure the learning gain. Having this target document will make user knowledge gain estimations more concrete. This can also be viewed as a reference knowledge state that represents the level of knowledge that the user should reach to consider the learning task as achieved.

4.3.2 RULK components

Our proposed framework is composed of three main components : the **Feature Extractor** (γ), the **Updater** (σ), and the **Estimator** (θ). These components interact with each other in multiple situations. In this subsection, we present each of the components that constitute the RULK framework. The details of three possible instantiations of RULK, namely RULK_{KW}, RULK_{LM}, and RULK_{NE}, and how they implement each component are provided in Section 4.5.2. We show an overview of RULK in Figure 4.1.

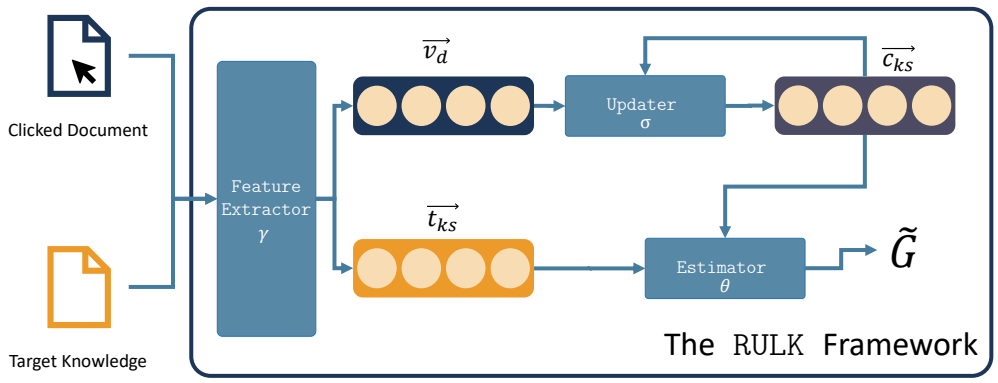

Figure 4.1 – The RULK framework and its main components. First, a clicked document d is transformed into $\overrightarrow{v_d}$ by γ. Next, σ updates the current state $\overrightarrow{c_{ks}}$ with $\overrightarrow{v_d}$. Finally, θ compares $\overrightarrow{c_{ks}}$ to a target knowledge vector $\overrightarrow{t_{ks}}$ to get an estimation of the user's knowledge gain in the session ($\widetilde{G}$).

4.3.2.1 Feature Extractor (γ)

The Feature Extractor γ is responsible for extracting meaningful features from any given piece of text t using various natural language processing techniques. These features are then represented as a fixed-size vector $\overrightarrow{v_t}$ with a dimension of m.

$$\overrightarrow{v_t} = \gamma(t). \tag{4.1}$$

By maintaining a consistent vector size of m and encoding all documents in the same embedding space, the feature extractor γ facilitates easy comparison and combination of documents within the framework RULK. This is achieved by representing documents as fixed-size vectors in a shared embedding space, allowing for meaningful comparisons. Consequently, the distance between two documents in this space can accurately reflect their semantic similarity. That means that by applying Equation 4.1, a visited document d will be encoded into $\overrightarrow{v_d}$ which will be fed to the Updater σ as an input to add it to the user's current state.

We will represent the additional knowledge (delta target knowledge) that needs to be incorporated into the user's current state. Also, the target document will be encoded as the target knowledge state $\overrightarrow{t_{ks}}$, which represents the learning objective and will be used later as input by the Estimator θ to estimate the knowledge gain.

4.3.2.2 Updater (σ)

To track the user's knowledge state, we follow Brooke's (Brookes, 1980) by considering that the process of learning is an "update" of a current state to a new one as described by the fundamental equation :

$$K[S] + \Delta I = K[S + \Delta I] \tag{4.2}$$

where $K[S]$ is the user's current knowledge representation, the additional information source ΔI is the new arriving information, and $K[S + \Delta I]$ is the revised new user knowledge.

RULK tracks the user's knowledge through an internal state represented by a vector *current knowledge state* $\overrightarrow{c_{ks}}$ having the same length m as the $\overrightarrow{v_d}$ embeddings produced by γ. σ updates $\overrightarrow{c_{ks}}$ with the new information $\overrightarrow{v_d}$ coming from a visited document. The new knowledge state of the

user is $\overrightarrow{c\prime_{ks}}$, also said the revised state of the user's knowledge $\overrightarrow{c_{ks}}$ after reading a document d. We assume that users were able to absorb the content of that document :

$$\overrightarrow{c\prime_{ks}} = \sigma(\overrightarrow{c_{ks}}, \overrightarrow{v_d}) = \overrightarrow{c_{ks}} + \overrightarrow{v_d}, \tag{4.3}$$

where the document is represented by $\overrightarrow{v_d}$, generated by Equation 4.1 ; and σ is a function that takes $\overrightarrow{c_{ks}}$ and $\overrightarrow{v_d}$ and combines them into an updated representation of the users' knowledge.

4.3.2.3 Estimator (θ)

RULK *estimates* the users' knowledge gain $\widetilde{G}$ on the topic during the session by comparing the user's current knowledge state, $\overrightarrow{c_{ks}}$ to the target $\overrightarrow{t_{ks}}$ using some similarity measure θ :

$$\widetilde{G} = \theta(\overrightarrow{c_{ks}}, \overrightarrow{t_{ks}}), \tag{4.4}$$

The intuition behind θ is that the user, by progressing in their session, "moves" their knowledge state ($\overrightarrow{c_{ks}}$) towards the target ($\overrightarrow{t_{ks}}$). As both vectors are in the same embedding space, we interpret the similarity between $\overrightarrow{c_{ks}}$ and $\overrightarrow{t_{ks}}$ as an estimate of how close the user is to acquire the knowledge contained in $\overrightarrow{t_{ks}}$.

It is important to note that while some users may have the willingness to complete their learning objective and reach the entirely of target document of $\overrightarrow{t_{ks}}$, others may stop before the defined target. Some users for example may be satisfied with acquiring only a portion of the target.

4.4 Exploring different structures for the knowledge state

Vector space models are commonly employed in personalization and recommendation frameworks, where both the user and the recommended items are represented as vectors. Several document personalization and recommendation frameworks used simple yet very common vector space models to represent the user and the recommended items, like the Tf-idf retrieval for example (Pazzani & Billsus, 2007 ; Van Meteren & Van Someren, 2000 ; Castro, Rodriguez, & Barranco, 2014). In this section, we explore various structures for our RULK framework, to represent the user, the recommended item (document), and the learning goal within a vector space model.

4.4.1 Vocabulary learning model

Vocabulary learning is a fundamental aspect of knowledge acquisition that includes the process of acquiring and mastering the terms of the vocabulary, their meanings, and their usage. Reading is a powerful tool for vocabulary learning as it exposes individuals to a wide range of words and their usage in context. In our work, we will track the user's vocabulary knowledge by counting the words they read. To achieve a vocabulary learning goal, a specific measure of progress toward the goal must be established. We will set this measure to be the target number of words to be read. The progress towards the goal will be assessed by comparing the number of occurrences of the keyword that the user has read to the target number.

We formalize our vocabulary learning model as follows :

— Let T be the topic the user is learning represented by a set of pieces of information to be learned.

— $K_T = \{kw_1, kw_2, \ldots, kw_m\}$ be the set of target **keywords** needed to learn the topic T. The set can be defined by an expert model, or extracted from a reference target document using automatic keyword extraction methods.
— $\overrightarrow{t_{ks}} = \{tks_1, tks_2, \ldots, tks_m\}$ be the **t**arget **k**nowledge **s**tate vector. tks_i denotes the desired **frequency** with which the user should read the i-th keyword kw_i in the set of target keywords K_T.
— $\overrightarrow{c_{ks}} = \{cks_1, cks_2, \ldots, cks_m\}$ **c**urrent **k**nowledge **s**tate vector. The variable cks_i represents the **frequency** with which the user has encountered the i-th keyword, denoted as kw_i, within the set of target keywords K_T.
— $\overrightarrow{v_d} = \{dko_1, \ldots, dko_m\}$ is the representation of a document where dko_i is the **frequency** of the i-th keyword kw_i in the document.

Both $\overrightarrow{t_{ks}}$ and $\overrightarrow{c_{ks}}$ have the same length m. We adopt the assumption made in (Syed & Collins-Thompson, 2017a) that reading an instance of keyword tk_i monotonically increases the user's knowledge of that keyword. Each time a user reads an occurrence of a word, the system adds *one* to the related vocabulary learning count. As we focus on short-term learning outcomes for a search session, we can make a simplifying assumption that minimizes concerns about memory decay over time. As a consequence, the occurrence count of the read document will not decrease during the session.

4.4.2 Language models

Language Models (LM) are computational models that are trained to predict the probability of the next word in a sequence of text given the previous words. They are a fundamental component of many natural language processing tasks, such as machine translation, speech recognition, text generation, and sentiment analysis, and have been commonly used in the information retrieval field (Ponte & Croft, 2017 ; Zhai & Lafferty, 2004). They can be trained on a variety of text sources, such as large corpora of general text, specialized domain-specific texts, or even individual users' text data. Although language models have been used in information retrieval for a long time in the literature, most of the applications have used them to calculate the similarity between a document and a query (Lafferty & Zhai, 2001), without taking into account the user context and profile. Zamani and Shakery used statistical language models for a content-based filtering system and profiled the user using a simple yet common unigram language model (Zamani & Shakery, 2018). Transformers-based language models (Vaswani et al., 2017), particularly those based on BERT (Devlin, Chang, Lee, & Toutanova, 2019), have been shown to excel in multiple tasks (Lin, Nogueira, & Yates, 2020), even if not fine-tuned for a specific domain (Thakur, Reimers, Rücklé, Srivastava, & Gurevych, 2021 ; Reimers & Gurevych, 2019). Many recent work have shown the exceptional ability of Large Language Models (LLM) to capture the semantic meaning of texts (Tamkin, Brundage, Clark, & Ganguli, 2021 ; Bogers & Van den Bosch, 2007). However, there is little work on profiling the users using those methods, and no work about profiling their knowledge and needs. Given their success, we propose an approach that leverages the natural language processing capabilities of LLMs and transformer-based models to represent both the text that brings new information to users (visited documents) and the set of information that they have an objective to learn(target document).

4.4.3 Knowledge graphs and named entities

4.4.3.1 Background about knowledge graphs

Knowledge Graphs (KGs) are a type of graph database that captures complex relationships between entities in a knowledge domain. They are used to represent knowledge in a machine-readable format, allowing for efficient storage and retrieval of information. KGs are becoming increasingly popular in various domains, including information retrieval. In IR, KGs are used to enhance search results by providing additional information and context about entities and their relationships.

In this chapter, we will assume that knowledge graphs follow a standard and technical infrastructure like the ones provided by the W3C for the Semantic Web, i.e. : OWL (the Web ontology language, based on description logics) based on the underlying data model RDF (the resource definition standard). This would allow a practical implementation of our proposal using state-of-the-art knowledge engineering technologies.

The basic statement of RDF is a *triple* $\langle s, p, o \rangle$, where s is called the subject, p the predicate, and o the object. The subject of a triple is a *resource* (or, in other words, an *entity*), represented by an internationalized resource identifier (*IRI*), like `<http://example.org/resource/LHR>`; the (binary) predicate represents a *property* of the subject, denoted by an IRI, like `<http://example.org/ontology#city>` or `<http://example.org/ontology#iataCode>`; the object, which represents a value of that property, may be a resource, denoted by an IRI, like `<http://example.org/resource/London>`, or a data value, such as a string, a number, or a date, denoted by a *literal*, like `"LHR"`, `2`, or `07:30`. In addition, the subject and object can be so-called *blank nodes*, which correspond to anonymous resources and can be understood as a kind of existentially quantified variables. For the sake of readability and conciseness, when several IRIs share the same base, a prefix may be defined, for instance

```
@prefix & : & <http://example.org/resource/> .
@prefix & o: & <http://example.org/ontology#> .
```

and the IRIs may then be abbreviated as `:LHR` instead of the full `<http://example.org/resource/LHR>` or `o:iataCode` instead of the full `<http://example.org/ontology#iataCode>`.

An important thing to observe is that, behind an IRI, which is essentially an identifier, many different notions can hide, like an instance (i.e., a constant, what is called an *individual name* in description logics), a binary predicate (i.e., a binary relation, what is called a *role* in description logics and a *property* in OWL), or a concept (i.e., a unary predicate, called a *class* in OWL). It is exactly this uniform naming convention that makes RDF so flexible and versatile. Thus, for instance, an assertion of the form $C(a)$, where C is a unary predicate (a concept) and a is the name of an individual, may be represented as an RDF triple $\langle a, \mathsf{rdf:type}, C \rangle$, which may be paraphrased as "a is an instance of C", thanks to the rdf :type relation, and an assertion of the form $R(a, b)$, where R is a binary predicate and a and b are entity names, may be represented as an RDF triple $\langle a, R, b \rangle$, for example

```
:LHR o:city :London .
```

which may be paraphrased as "the city of the Heathrow Airport is London".

A collection of RDF triples, which may represent assertions and other OWL axioms with a uniform syntax, may be regarded as a knowledge base under the open-world assumption, from

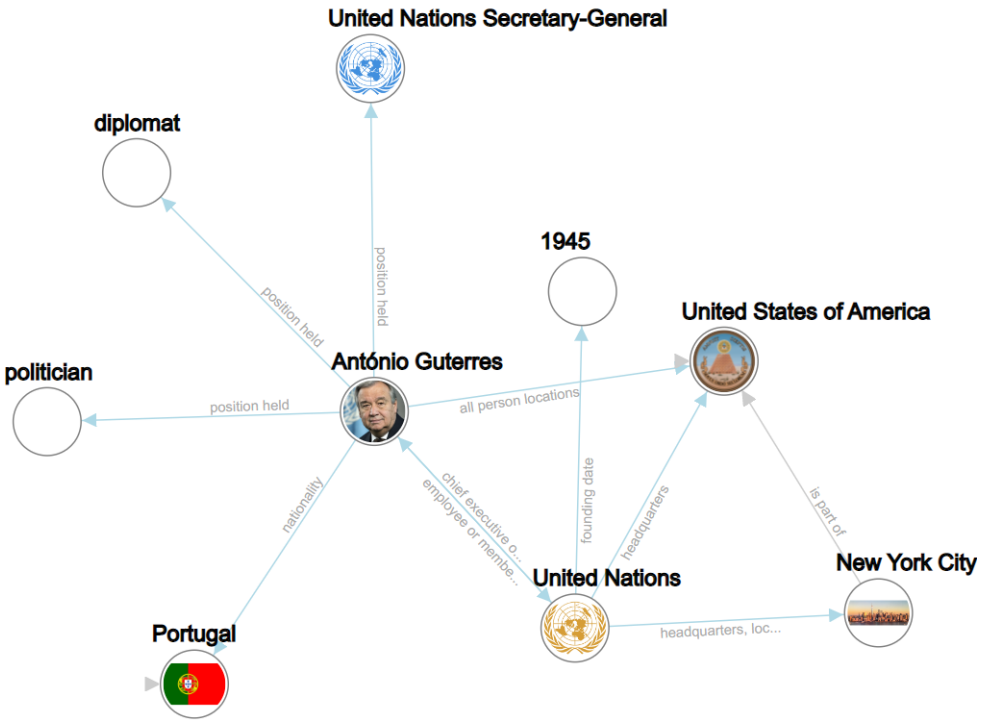

Figure 4.2 – Knowledge graph representation of information extracted from a sample text.

which other triples can be deduced using an inference engine called an OWL *reasoner*. * Furthermore, a collection of RDF triples intrinsically represents a directed multi-graph (an RDF graph), whose vertices are resources ; every triple $\langle s, p, o \rangle$ then represents an arc of type p from vertex s to vertex o.

4.4.3.2 Personal knowledge graphs

Personal Knowledge Graphs (PKG) are small-scale knowledge graphs that contain user-specific data, preferences, and interests. They are built on top of existing knowledge graphs, such as DBpedia or Wikidata, and are personalized to the user's context. PKGs can be used to enhance information retrieval and recommendation systems by providing more relevant and personalized results to the user.

The concept of personal knowledge graphs was recently introduced to offer pocket-sized knowledge graphs related to users' interests (Safavi et al., 2019 ; Daoud et al., 2009 ; Daoud, Tamine, & Boughanem, 2010), user's personal information (Balog & Kenter, 2019), life events (Yen, Huang, & Chen, 2019). Recently, Ilkou *et al.* proposed a PKG framework specifically designed for educational and learning purposes within e-learning platforms (Ilkou, 2022). The framework captures the user's interests and builds a graph that can be used for personalized learning. While the field of personal knowledge graphs is a growing area and is currently gaining more attention, to our knowledge there are currently no personal knowledge graphs that represent the user's knowledge using a knowledge graph.

∗. Examples of popular OWL reasoners are Fact++, HermiT, and Pellet

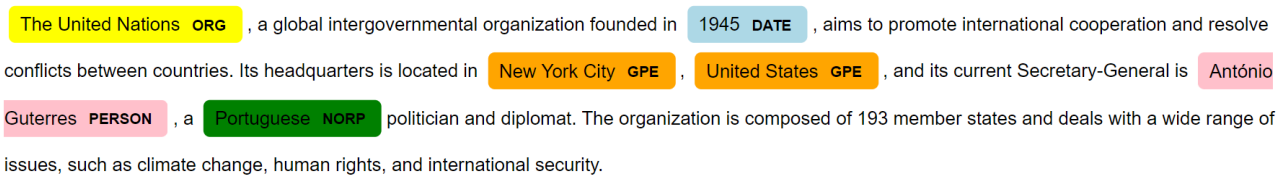

Figure 4.3 – Named entity recognition task for a sample text.

Our ultimate objective is to represent the user's knowledge in the form of a knowledge graph, where entities are interconnected and linked by verbs and states. Ideally, we also aim to represent the target knowledge as a graph. To assess the user's learning outcome, we will compare and quantify the similarity between both knowledge graphs. Figure 4.2 illustrates a knowledge graph created from a sample text using the Diffbot Natural Language API (Diffbot, s. d.). This graph could potentially represent the target knowledge extracted from a reference document.

4.4.3.3 Named entities

Knowledge graphs are frequently used to store interlinked descriptions of entities such as objects, events, situations, or abstract concepts, while also encoding the semantics underlying the used terminology. As a first step towards achieving our goal, which was mentioned in the previous section of representing the user's knowledge with a knowledge graph, we aim to recognize the entities and explore whether they will improve our previous vocabulary-keyword models or not. The next step, which is left for future work, involves linking these entities with relationships to represent the knowledge states and construct a knowledge graph (KG). This will enhance the understanding of the representations by providing a more comprehensive framework.

Named entities are real-world objects or concepts that have a specific name or label, such as people, organizations, locations, and events. They are commonly used as nodes in knowledge graphs. Each named entity can be represented as a node in the graph, with properties or attributes attached to it that describe the entity, such as its type, name, and other relevant information. The task of identifying and classifying these named entities in text is called Named Entity Recognition.

We see the potential benefits of integrating named entities into the keyword vocabulary model. By incorporating named entities as standalone components, we aim to test the hypothesis that this could enrich the keyword model and enhance the system's performance. Named entities are particularly interesting as they typically carry meanings and are associated with standardized resources. For example, a named entity like "New York City" has a specific semantic meaning as a geographical location and is linked to standardized resources such as geographical databases or gazetteers. By leveraging this semantic meaning and standardized resources, we anticipated that incorporating named entities could enrich the keyword vocabulary model, making it more robust and capable of capturing the nuances of user queries.

We show a visualization of named entity recognition of a sample text in Figure 4.3.

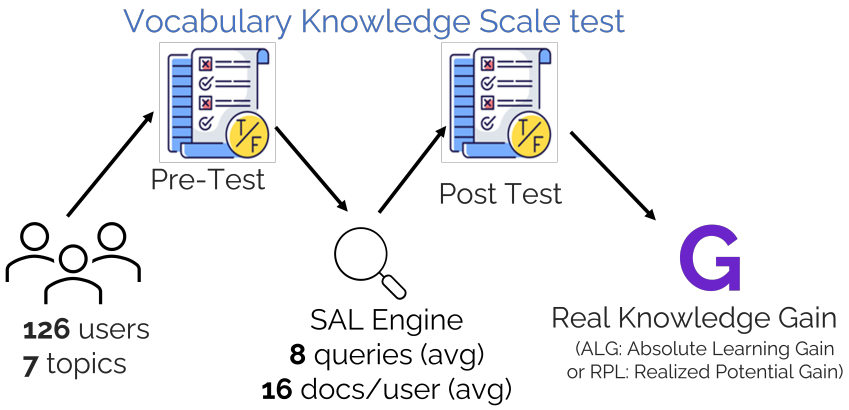

Figure 4.4 – Workflow of the experiment for measuring user's knowledge gain through a search as learning engine.

4.5 Experimental design

In this section, we discuss how we validated the three implementations of RULK : RULK_{KW} based on the vocabulary learning model discussed in Section 4.4.1, RULK_{LM} † discussed in Section 4.4.2 and RULK_{NE} ‡ discussed in Section 4.4.3.3. We begin by describing the dataset we adopted, which consists of real user logs from a publicly available dataset that contains the real user gain, G. Then, we explain how we applied the RULK framework to estimate the user's knowledge gain, $\tilde{G}$, and finally compared it to the real gain.

4.5.1 Dataset

To analyse how our implementations of RULK perform, especially when estimating the users' knowledge gain in a search session, we test them on a publicly available dataset of SAL sessions built with the logs from the study by (Câmara et al., 2021b) §.

4.5.1.1 Experiment topic selection

The authors of the dataset selected 7 out of 117 training topics from the TREC CAR 2017 (Dietz, Gamari, Dalton, & Craswell, 2018) dataset. The CAR dataset was originally designed to find relevant passages for Wikipedia headings and provide a hierarchical structure of topics. The authors selected topics that had at least two hierarchical levels (between 11 and 27 subtopics) and that were assessed as complex and difficult by 17 STEM (Science, Technology, Engineering, and Mathematics) students. The selection of the most difficult topics was justified by maximizing the potential learning of the participants during the experiment and ensuring that the knowledge gained would be considerable. The resulting seven topics were : Business cycle, Ethics, Genetically

†. RULK_{KW} and RULK_{LM} implementations can be found at https://github.com/ArthurCamara/RULK_SAL

‡. RULK_{NE} implementation can be found https://github.com/dimaelzein/RULK_entities

§. The data is available at https://github.com/ArthurCamara/CHIIR21-SAL-Scaffolding

	Total	Mean	Median
Number of users per topic	126	18.14 ± 2.79	19.0
Number of topics	7	-	-
Number of queries	1095	8.62 ± 6.47	7.0
Number of documents clicked	2116	16.66 ± 8.85	16.0
Number of snippets seen	15184	119.56 ± 72.43	105.0
Documents Clicked per query	-	2.78 ± 2.50	2.11
Session duration (minutes)	-	56.18 ± 14.58	54.05
Document dwell time (seconds)	-	79.94 ± 69.77	60.0
Pre-test scores (vks^{pre})	-	1.07 ± 1.60	0.00
Post-test scores (vks^{post})	-	6.21 ± 4.09	6.00
Actual Learning Gain (ALG)	-	0.53 ± 0.38	0.50
Realised Potential Learning (RPL)	-	0.28 ± 0.20	0.25

TABLE 4.1 – Statistics, per user, extracted from the dataset used by Camara2021SearchingTL.

modified organisms, Irritable bowel syndrome, Noise-induced hearing loss, Radiocarbon dating considerations, and the Subprime mortgage crisis.

At the start of the study, the system measured the previous knowledge of each participant on two randomly selected topics and selected the topic the user demonstrated a lower knowledge of as their target topic T.

4.5.1.2 Participants and recruitment

The study recruited participants through the *Prolific Academic platform* ¶, with eligibility criteria including a minimum of 15 previous submissions, an approval rate of over 90%, and being a native English speaker. Participants were compensated with £6 per hour for the one-hour study. The final dataset consisted of 126 valid participants, with 65 male, 59 female, and 2 participants choosing not to disclose their gender. The median age of the participants was 27, with a range from 18 to 63 years old. Of the participants, 44 reported having a high school degree as their highest formal education level, 47 had a Bachelor's degree, and 20 had a Master's degree, while the remaining 15 indicated other educational levels.

4.5.1.3 Real knowledge measurement

The dataset contains self-reported of users' knowledge before and after the search session, vks^{pre} and vks^{post} respectively. The knowledge were measured with a *Vocabulary Knowledge Scale (VKS)* test (Wesche & Paribakht, 1996; Stahl & Bravo, 2010), a commonly used method to measure user knowledge (Salimzadeh, Gadiraju, Hauff, & van Deursen, 2022; Roy et al., 2020; O'Brien, Kampen, Cole, & Brennan, 2020b; Syed & Collins-Thompson, 2017a). To prevent any potential influence on search behavior, the participants were not informed that the post-session tests would be the same as the pre-test.

¶. https ://www.prolific.co/

The VKS tests for each topic consisted of a set of 10 vocabulary concepts. Users were required to assess their knowledge levels for each concept. The selection of the tested concept terms followed a two-step process. Firstly, a list of 100 candidate unigram/bigram concepts was automatically extracted from the corresponding Wikipedia article of each topic, using the highest IDF score. Secondly, the authors manually selected 10 concepts from this list to be included in the tests. To self-assess the user's knowledge the users were presented with a 4-point scale questionnaire, asking about their familiarity with the ten topic-related selected vocabulary. The VKS test for a concept included using the following options :

1. I don't remember having seen this term/phrase before.
2. I have seen this term/phrase before, but I don't think I know what it means.
3. I have seen this term/phrase before and I think it means ...
4. I know this term/phrase. It means ...

If users self-assessed their knowledge of the vocabulary term with (3) or (4), they were requested to provide a definition of it in their own words.

To measure and quantify the user's familiarity with a vocabulary term v_i, score $vks(v_i)$ ranging between 0 and 2 were calculated based on the user's self-reported level as follows :

— Self-reported as level 1 or 2 means that the user does not know the term, $vks(v_i) = 0$.
— Self-reported as level 3 means that the user partially knows the term, $vks(v_i) = 1$.
— Self-reported as level 4 means the user fully understands the term, $vks(v_i) = 2$.

To measure the user's learning gain during their session, the difference between vks^{pre} and vks^{post} is computed using the *Absolute Learning Gain* ALG and the *Realized Potential Learning* RPL as follows :

$$
\begin{aligned}
ALG &= \frac{1}{10}\sum_{i=1}^{10} max(0, vks^{post}(v_i) - vks^{pre}(v_i)) \\
MLG &= \frac{1}{10}\sum_{i=1}^{10} 2 - vks^{pre}(v_i) \\
RPL &= \frac{ALG}{MLG}
\end{aligned}
\tag{4.5}
$$

where $vks(v_i)$ is the score of the user for the i-th term. MLG is the *Maximum Learning Gain* reflecting the maximum amount of *new* knowledge a user can acquire, given what they already know, which is 2 if the pre-test score is 0 or 1 if the pre-test score is 1. RPL represents the fraction of knowledge the user acquired from the total knowledge they could obtain in their session. RPL normalizes ALG by the maximum possible learning potential. In the rest of this chapter, we refer ALG and RPL to *real* knowledge gain.

The median number of queries ranges from 5 to 9.5 ; 1260 VKS questions were collected by the end of the experiment out of which 100 were sampled for self-reported quality inspection. Fifty of these were from knowledge level 3 and another fifty were from level 4. The provided definitions were labeled as correct, partially correct, or incorrect by two annotators. It was found that around 25-28% of the vocabulary scores were correct, 64% were partially correct, and less than 10% were incorrect. Consequently, the self-assessment measures could then be considered reliable.

4.5.1.4 Search tool and logged interactions

After assigning the search topic, the participants were granted access to *SearchX* (Putra, Moraes, & Hauff, 2018), an open-source, modular search framework that offers quality control features for crowd-sourcing experiments and detailed search logs. The *Bing* search API was utilized to retrieve relevant documents for the query. To eliminate documents originating from Wikipedia or similar pages, 72 domains known to be Wikipedia clones were filtered out. Thus, none of the documents presented to the users in the experimental search results were from Wikipedia or similar sites. The participants were instructed to spend at least 30 minutes gathering information about the given topic. The logged interactions included behavioral features, queries issued, and documents clicked on. Table 4.1 shows some statistics about the user's behaviour during the search session.

As the dataset does not contain the textual contents of the 1107 unique clicked documents, we used the Wayback Machine API ‖ to fetch the documents when the study was conducted (August 2020). Of all the documents, 33 did not have a snapshot available and were discarded from our experiments (i.e., we do not consider their impact on users' knowledge).

4.5.2 Methodology

We propose three possible implementations for deploying RULK in a search system. The first, called RULK_{KW}, relies on keyword-based feature extraction and is based on the vocabulary learning model, while the second, RULK_{LM}, uses LLMs, like BERT, and the third uses named entities RULK_{NE}. The main difference between the implementations is in the choice of the semantic space for representing documents, the user's knowledge, and the target document. To initialize the users' knowledge in the given context, we set the current knowledge vector ($\overrightarrow{c_{ks}}$) for all users to zero. We make the assumption that the users had no prior knowledge, especially considering that the experiment setup assigned them to the topic they had the least amount of knowledge. The Updater module σ in all three implementations estimates the user's knowledge gain using cosine similarity.

4.5.2.1 Target knowledge

The topics used in the original user study came from the list of topics used in the CAR track from TREC 2018 (Dietz et al., 2018). In that track, each topic is the title of a Wikipedia article from a 2018 dump. Therefore, all three of our implementations use these Wikipedia texts, from the same 2018 dump as the original paper, as "reference documents" for generating $\overrightarrow{t_{ks}}$. Furthermore, we use the same dump as in the original paper.

4.5.2.2 RULK_{KW} implementation : A Keyword-based Variant

This variant of the framework utilizes the vocabulary learner model discussed in Section 4.4.1. It assumes that the user's learning goal is to read a specific number of occurrences of topic-related keywords denoted as K_T. The set of target occurrences is obtained by feeding the target document into the γ feature. Similarly, document d is embedded in γ by counting the occurrences of keywords from K_T within it. As for the Updater module σ, it updates the user's knowledge representation $\overrightarrow{c_{ks}}$ as users read documents represented by $\overrightarrow{v_d}$, by incorporating the count of keywords from K_T found in the document, as shown in Equation 4.3.

‖. https://web.archive.org/

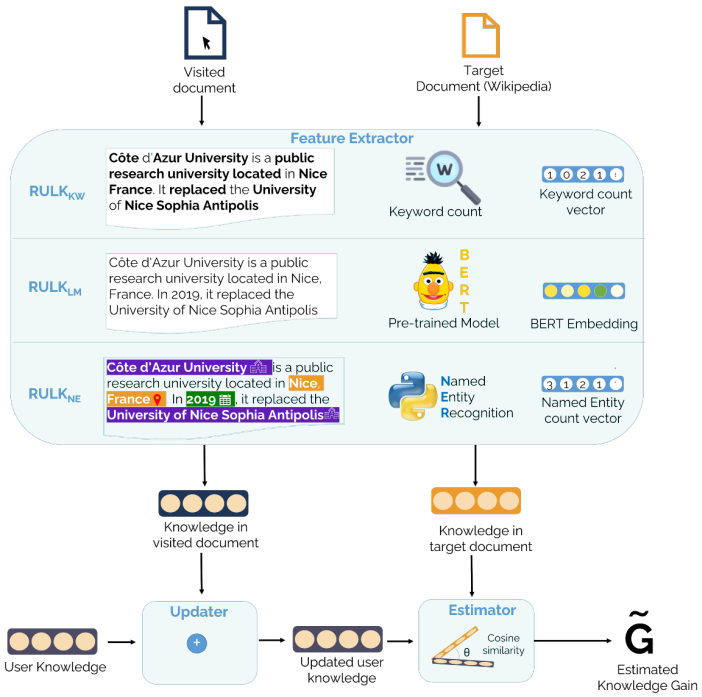

Figure 4.5 – Illustrating the three proposed implementations of the RULK framework, which consists of the Feature Extractor (γ), the Updater (σ), and the Estimator (θ).

The Feature Extractor γ in this implementation uses the *Yet Another Keyword Extraction (YAKE)* method (Campos et al., 2018). This method is a lightweight, unsupervised, automatic keyword extraction technique that relies on statistical features extracted from documents to select the most important keywords in a text. We set the maximum n-gram size to 3 ; however, we noticed that all the keyphrases extracted by *YAKE*, for all topics, were 1-gram. We also chose a value of $m = 10$ as the size of $\overrightarrow{t_{ks}}$, $\overrightarrow{c_{ks}}$, and $\overrightarrow{v_d}$. To avoid selecting excessively similar keywords, we implement the Porter Stemmer from the NLTK Python library (Bird, Klein, & Loper, 2009) to stem the keywords that are extracted by YAKE. Knowing that this stemming approach might generate duplicate keywords (e.g., "water" and "waters" having the same stem of "water"), we removed duplicated stems and replace them with the next most relevant keyword until we have a list of ten distinctive keywords per topic.

Upon the arrival of new information represented as $\overrightarrow{v_d}$, the Updater module (σ) adds the count vectors together using simple vector addition operation.

4.5.2.3 RULK$_{LM}$ implementation : language model-based variant

This variant implements the language model approach discussed in Section 4.4.2 to assess whether such models can capture features of the user's knowledge and predict the gain at any time of the session. Thus, we implement a BERT-based variant of the framework inspired by the method proposed by (Câmara et al., 2021b) to track user exploration of a topic. Both the target knowledge $\overrightarrow{t_{ks}}$ and clicked document's embedding $\overrightarrow{v_d}$ are represented by an embedding of fixed length $m = 384$, as generated by the same language model. Given a text (a visited d or, conversely, a target document) with k sentences $\{s_1, s_2 \dots s_k\}$, γ generates, for each sentence s_i,

an embedding of size m given by :

$$\overrightarrow{v_{s_i}} = BERT([CLS]; s_{i:l}; [SEP]), \tag{4.6}$$

where ; is a concatenation, l the maximum input size of the model, and $[CLS]$ and $[SEP]$ are special BERT tokens. $\overrightarrow{v_t}$ ($\overrightarrow{v_d}$ or conversely, $\overrightarrow{t_{ks}}$) is then given by an element-wise sum over all $\overrightarrow{v_{s_i}}$ as shown in Equation 4.7.

$$\overrightarrow{v_t} = \gamma(t) = \sum_{i=1}^{k} \overrightarrow{v_{s_i}} \tag{4.7}$$

Specifically, we use a MiniLM (Wang et al., 2020) model with 6 layers and a hidden layer's dimension of 384. The model was also fine-tuned on the MsMarco dataset (Craswell, Mitra, Yilmaz, Campos, & Lin, 2021), as made available in the SBERT framework (Reimers & Gurevych, 2019) **.

We split the documents into sentences using the *NLTK*'s implementation of the *Punkt Sentence Tokenizer*, feed each sentence individually into the γ, and sum their respective embeddings. We show in Figure 4.6 the workflow of the text embedding done by γ.

The Updater σ performs then a simple element-wise sum over all elements of $\overrightarrow{v_d}$ and $\overrightarrow{c_{ks}}$, to integrate the new information acquired by the user from the document. As $\overrightarrow{t_{ks}}$ and $\overrightarrow{c_{ks}}$ are vectors in the same embedding space, θ, similarly to $\texttt{RULK}_{\texttt{KW}}$, is also the cosine similarity between $\overrightarrow{t_{ks}}$ and $\overrightarrow{c_{ks}}$, as shown in Equation 4.8. We would like to mention that we also experimented with other options for σ (e.g. averaging the embeddings instead of summing) and θ (e.g. using Euclidean distance on a normalized vector), and all options resulted in very similar results.

4.5.2.4 $\texttt{RULK}_{\texttt{NE}}$ implementation : entity-based variant

The $\texttt{RULK}_{\texttt{NE}}$ implements the structure proposed in Section 4.4.3.3. The Feature Extractor γ here produces a collection of links to knowledge graph resources, corresponding to named entities. To produce those links from a document, and after setting a reference knowledge graph as background knowledge, two NLP tasks should take place :

1. named entity recognition (i.e., spotting chunks of text that are likely to refer to specific entities such as people, places, organizations, etc.), and
2. entity linking (i.e., establishing a link between a recognized named entity and a resource defined in the reference knowledge graph).

While these two tasks can be challenging, a multitude of tools that solve them with acceptable performance, albeit not always perfectly, has become available in recent years, making it possible to envisage what we are proposing. One can only foresee that these tools will be improved and new, even better tools will become available in the near future, thus potentially contributing to making $\texttt{RULK}_{\texttt{NE}}$ more and more accurate. The reference knowledge graph can be any of the large general-purpose RDF datasets available in the Linked Open Data cloud, like DBpedia, Yago, or Wikidata. For the entity recognition task, we detect the *named entities* using the *Spacy* (Honnibal & Montani, 2017) Python library. We chose the English pipeline *en_core_web_sm*, which is trained on written Web text (blogs, news, comments). Then automatically link the entities by annotating the texts with DBpedia resources using the *dbpedia_spotlight* tool (Mendes, Jakob, García-Silva,

**. https://huggingface.co/sentence-transformers/msmarco-MiniLM-L6-cos-v5

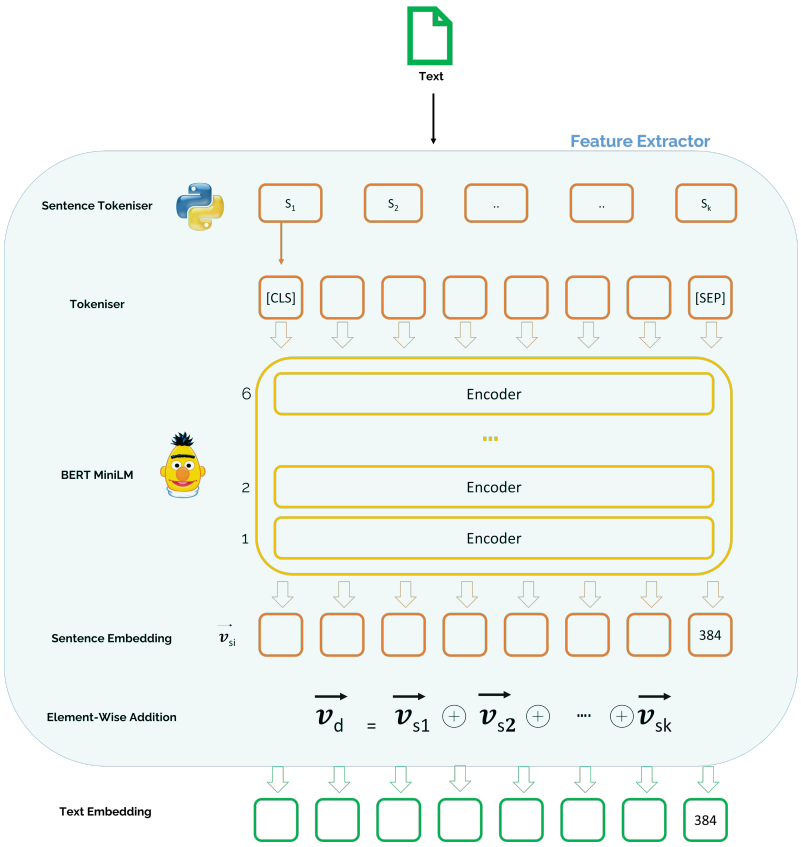

Figure 4.6 – Text Embedding by Feature Extractor Module of γ of $\texttt{RULK}_{\texttt{LM}}$.

& Bizer, 2011). The Feature Extractor γ module encodes extracts the set of target keywords K_T as the 10 most common entities in the target document and sets the target knowledge as a vector $\overrightarrow{t_{ks}}$ of the counts of the occurrences of these keywords in the target document. The same module also encodes a read document as a vector $\overrightarrow{v_d}$ of the counts of the keywords K_T in it.

The Updater σ is then an addition operation of the two count vectors $\overrightarrow{c_{ks}}$ and $\overrightarrow{v_d}$.

4.5.2.5 Similarity calculation and comparison of results

The three implementations share the same Estimator (θ), implemented as a cosine similarity between $\overrightarrow{t_{ks}}$ and $\overrightarrow{c_{ks}}$:

$$\widetilde{G} \approx \theta(\overrightarrow{c_{ks}}, \overrightarrow{t_{ks}}) = \frac{\overrightarrow{c_{ks}} \cdot \overrightarrow{t_{ks}}}{|\overrightarrow{t_{ks}}|\ |\overrightarrow{c_{ks}}|}. \tag{4.8}$$

As discussed in Section 4.4, the user's estimated knowledge gain $\widetilde{G}$ is calculated as the cosine similarity between the tracked knowledge $\overrightarrow{c_{ks}}$ and the target knowledge $\overrightarrow{t_{ks}}$. To address the **[RQ4-1]** and assess the validity of our framework, we measure the correlation between the estimated gain $\widetilde{G}$ and the real user's knowledge G (*ALG* and *RPL*).

4.5.2.6 `RULK` mixed approaches

To address **[RQ4-2]** we extend our framework (`RULK`) to include a set of combination between the three proposed implementations. Our hypothesis is that combining the three models based on keywords, language models, and named entities can result in a more comprehensive and accurate representation of the user's knowledge compared to using each model individually. By combining these models, we assume that the strengths of each implementation will be leveraged to create a more comprehensive and accurate representation of the user's knowledge. The combination of these models provides a more holistic understanding of the user's knowledge by capturing diverse aspects of the text.

We assume that these three approaches will complement each other in capturing different characteristics of the represented knowledge. The keyword-based $\texttt{RULK}_{\texttt{KW}}$ could be useful for capturing important terms and phrases directly extracted from the text of the visited document, providing insights about the knowledge acquired from the documents. The language model-based $\texttt{RULK}_{\texttt{LM}}$ could be powerful in capturing contextualized representations of words and phrases, capturing syntactic and semantic relationships between words and nuances of meaning. Lastly, the named entity-based $\texttt{RULK}_{\texttt{NE}}$ could provide valuable information about the specific entities mentioned in the text and their relationships, relevant for understanding the user's knowledge in a particular domain or topic.

On the other side, combining these models can potentially overcome their individual limitations. For example, BERT methods can cover the lack of contextual nuances of the keywords. The keyword-based model can capture the main topics covered in the text, the BERT language model can capture contextualized representations of words and phrases, and the named entity-based model can capture specific entities mentioned in the text. In such combinations, each instantiation has the potential to contribute to the overall estimation of the knowledge gain by capturing some specific characteristics of the texts of the framework.

We implement the mixed approach defined by an interpolated estimator θ, parameterized by α and β, defined as follows :

Method	ALG	RPL
$\texttt{RULK}_{\texttt{KW}}$	0.3022	0.3086
$\texttt{RULK}_{\texttt{LM}}$	0.2955	0.2923
$\texttt{RULK}_{\texttt{NE}}$	0.0931	0.1185
$\texttt{RULK}_{\texttt{KW+LM}}$	**0.3164**	**0.3192**
$\texttt{RULK}_{\texttt{NE+KW}}$	0.3184	0.3333
$\texttt{RULK}_{\texttt{NE+LM}}$	0.3228	0.3309
$\texttt{RULK}_{\texttt{NE+LM+KW}}$	**0.3378**	**0.3490**

TABLE 4.2 – Pearson's correlation between a given implementation of the framework and real user's learning. **bold** values indicate the best correlation against a learning metric.

$$\theta_{\texttt{RULK}_{\texttt{LM+KW+NE}}} = \alpha\widetilde{G}_{\texttt{RULK}_{\texttt{LM}}} + \beta\widetilde{G}_{\texttt{RULK}_{\texttt{KW}}} + (1-\alpha-\beta)\widetilde{G}_{\texttt{RULK}_{NE}} \tag{4.9}$$

where $\widetilde{G}_{\texttt{RULK}}$ is the estimated knowledge gain according to the respective RULK implementation.

4.6 Results

4.6.1 RULK estimation accuracy

To answer our first research question **[RQ4-1]**, we test the validity of the RULK by computing the Pearson's correlation between the *real* knowledge gain G of a user and the *estimated* knowledge gain $\widetilde{G}$, as measured by each implementation's θ.

4.6.1.1 $\texttt{RULK}_{\texttt{KW}}$ estimation accuracy

As shown in Table 4.2, on the set of all users, the correlation values for the estimated knowledge gain of $\texttt{RULK}_{\texttt{KW}}$ are 0.3022 and 0.3086 respectively with ALG and RPL, indicating a moderate positive correlation between the estimated knowledge gain using the keyword-based model and the real user's knowledge gain. Keywords in the $\texttt{RULK}_{\texttt{KW}}$ were selected based on their relevance to the target document, using the YAKE automatic extraction method. These keywords can provide direct insights into the key information in the target and can be an effective representation of it. The good selection of the target keywords K_T and their corresponding required occurrences may have contributed to having a good representation of the tracked knowledge, which has been reflected in the estimated gains as well.

4.6.1.2 $\texttt{RULK}_{\texttt{LM}}$ estimation accuracy

When compared to $\texttt{RULK}_{\texttt{KW}}$, the results of $\texttt{RULK}_{\texttt{LM}}$ show a slightly lower positive moderate correlation, with values of 0.2955 and 0.2923 against ALG and RPL respectively. Language models, as BERT used in $\texttt{RULK}_{\texttt{LM}}$, are designed to capture the representation of words and phrases by considering their context and surrounding text in both forward and backward directions while considering the semantic relationships between the words. The correlation results are not unexpected given that the language model approach, such as $\texttt{RULK}_{\texttt{LM}}$, is capable of capturing more

nuanced relationships between words and is expected to have acceptable performance. However, it was surprising to observe lower correlation results compared to the keyword model, as the language model is a more complex approach that is designed to capture semantic relationships between words.

In an attempt to enhance those results, we hypothesized some concerns about the effectiveness of the update process, given that the values in the language model vectors may not be easily interpretable and adding them together may not yield the desired results. We explored other common approaches such as truncating the document, averaging the sentences, and using the most similar sentence to the user's query (MaxP) (Z. A. Yilmaz, Wang, Yang, Zhang, & Lin, 2019). These alternative approaches showed worse or similar accuracy results compared to simply adding the values together. Therefore, the value addition had no impact on the performance of the method.

In general, the results suggest that both keyword-based and language model-based approaches performed effectively in estimating knowledge gain, likely due to their ability to capture different aspects of the text, such as important words, concepts, topics, and contextualized representations of words and phrases.

4.6.1.3 $\texttt{RULK}_{\texttt{NE}}$ estimation accuracy

The correlation values for $\texttt{RULK}_{\texttt{NE}}$ estimation are 0.0931 and 0.1185 with ALG and RPL, respectively, indicating a weak positive correlation between the estimated knowledge gain using the named entity-based model and the real user's knowledge gain.

The results of the named entity approach looked disappointing, suggesting that the named entity-based approach ($\texttt{RULK}_{\texttt{NE}}$) may not be as effective in accurately estimating the user's knowledge gain compared to the other two models ($\texttt{RULK}_{\texttt{KW}}$ and $\texttt{RULK}_{\texttt{LM}}$). The named entity-based approach may solely focus on capturing specific entities mentioned in the text and their relationships, which may not always be the most relevant or comprehensive representation of the user's knowledge gain, especially in certain texts where named entities alone might not be reflective of the subject topic. Additionally, the accuracy of named entity recognition can be influenced by various factors such as the quality and completeness of the named entity recognition system, the domain-specific nature of the data, and the diversity of named entities mentioned in the text, which may impact the performance of the named entity-based approach.

4.6.2 Combined RULK estimation accuracy

To answer our second research question **[RQ4-2]**, we combined different approaches as proposed in Equation 4.9. The accuracy of these mixed approaches is shown in the lower part of Table 4.2, which displays the correlation between the estimated knowledge gain and the real learning outcome.

It is clear that mixing the approaches enhanced the correlation with the real gain than when individual approaches. The most significant improvement was observed when combining all three approaches resulting in a correlation of 0.3378 and 0.3490 for ALG and RPL respectively. These results suggest that combining different models can leverage the strengths of each and result in a more comprehensive and accurate estimation of the user's knowledge gain.

While the results of NE alone looked disappointing, it is interesting to notice that all the combinations that include NE outperform all the ones that do not include it. For example, when NE was added to the LM, the correlation increased from 0.2955 to 0.322. This is clear evidence that

		$\text{RULK}_{\text{KW+LM}}$	$\text{RULK}_{\text{NE+LM}}$	$\text{RULK}_{\text{NE+KW}}$	$\text{RULK}_{\text{NE+LM+KW}}$
ALG	α	0.44	0.82	-	0.44
	β	0.66	-	0.86	0.41
	$1-\alpha-\beta$	-	0.18	0.14	0.150
RPL	α	0.38	0.80	-	0.39
	β	0.62	-	0.84	0.44
	$1-\alpha-\beta$	-	0.20	0.16	0.169

TABLE 4.3 – Comparing parameters with the different mixture models.

the proposed representation is complementary to the others, i.e., capable of capturing something that the others miss.

Indeed, the difference between keywords and entities such as those that are stored in a knowledge graph is that keywords in general correspond to individual words, whereas entities correspond to concepts or instances of concepts whose lexicalization may involve phrases. The word embedding produced by the language model too, like keywords, operates at the level of individual words, although, by taking context into account, it can be able to distinguish different meanings of the same word or merge different words in the same meaning. Entities, however, when they are successfully recognized and linked to background knowledge, are much more precise because they are capable of designating very specific concepts. Consider for example two texts like "the President of the United States was in Manchester" and "the President of Manchester United is in the States" : after eliminating the stop words, the two texts contain the same keywords (Manchester, President, States, United) ; in terms of entities, however, they differ : [President of the United States] and [United States] are in the former, while [Manchester United (football team)] is in the latter.

Table 4.3 presents the optimized parameters of the mixed models, which are tuned based on the ALG and RPL metrics, respectively. Interestingly, the combinations that include NE perform the best, but the weight of NE in these combinations is lower compared to the other two implementations, ranging from 14% to 20%. This implies that NE contributes less to the estimation of the user's knowledge gain. For example, the $\text{RULK}_{\text{NE+LM}}$ model shows a correlation of approximately 0.032, while the associated parameter for NE, $(1-\alpha-\beta)$, is 0.18 for ALG and 0.20 for RPL.

4.6.3 Impact of session length on RULK accuracy

We address here the third research question **[RQ4-3]**. The focus of this question is investigating how the accuracy of the representations employed in RULK is affected by different session lengths.

We split users into quarters based on the length of their sessions as given by three measurements : number of queries issued, number of documents clicked, and session duration in minutes. We used *quintiles* as a technique to split the data into four groups of equal size.

We show the results of this analysis in Figure 4.7.

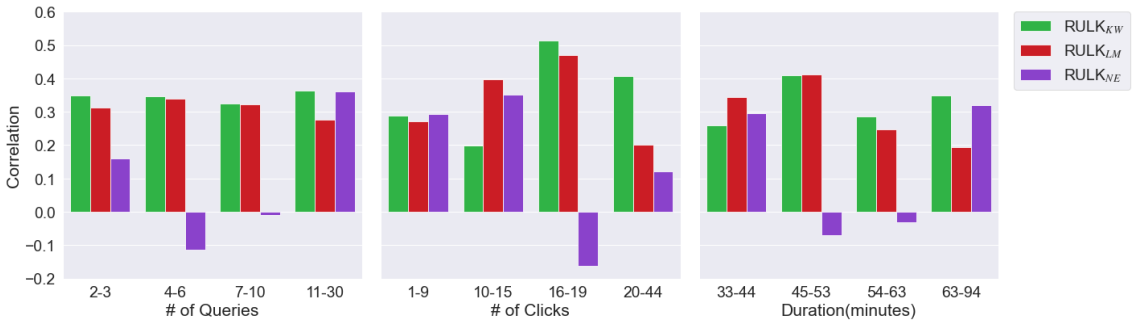

Figure 4.7 – Pearson's correlations between estimated and measured (RPL) knowledge gains by quintile.

4.6.3.1 Number of submitted queries

The analysis of Figure 4.7 reveals that the correlation between the estimated knowledge gain values by RULK$_{KW}$ and the real gain remains relatively constant across the different number of query ranges. The figure indicates a moderate positive correlation with a slight variation observed within the different studied percentile ranges. The highest reported correlation value was 0.364 observed in the percentile range of [11-30] queries, and the lowest correlation value of 0.325 was observed in the percentile range of [7-10] queries.

Similarly, the correlation results obtained RULK$_{LM}$ show a relatively constant pattern, with correlation values ranging between 0.276 and 0.339. As for the performance of RULK$_{NE}$, it shows no consistent pattern or trend of the correlation as the number of queries grows. The correlation values vary widely across different percentile ranges, ranging from positive values (e.g., 0.161 at the [2-3] percentile range) to negative values (e.g., -0.113 at the 4-6 percentile range and -0.009 at the 7-10 percentile range), and then back to positive values (e.g., 0.360 at the [11-30] percentile range).

In summary, the results suggest that the correlation between the knowledge gain estimation remains relatively constant across different query ranges for RULK$_{KW}$ and RULK$_{LM}$. However, RULK$_{NE}$ shows no consistent pattern or trend of correlation as the number of queries grows, with widely varying correlation values across different percentile ranges.

4.6.3.2 Number of visited documents

The number of visited documents could be the most representative indicator of session length and may have a significant impact on the performance of RULK estimations. This is because, as the user reads more documents, more content is extracted and added to the knowledge representations, causing the vectors of the knowledge representation specifically to be updated more and the values within them to increase. The figure shows a reliable performance for RULK$_{KW}$ ranging between 0.200 and 0.513 and also for RULK$_{LM}$ ranging from 0.2010 and 0.470 without showing a particular trend. The RULK$_{NE}$ showed a good correlation in the first two and the last quartiles with a correlation between 0.1218 and 0.2920 and a negative correlation of -0.1634 on the [16-19] quartile.

4.6.3.3 Session duration

For the session duration, we did not observe any specific pattern in the correlation resulting from $\texttt{RULK}_{\texttt{KW}}$ and $\texttt{RULK}_{\texttt{LM}}$. However, both showed a moderate positive correlation across all durations, with $\texttt{RULK}_{\texttt{KW}}$ ranging between 0.2287 and 0.40, and $\texttt{RULK}_{\texttt{LM}}$ ranging between 0.200 and 0.413.

On the other hand, for $\texttt{RULK}_{\texttt{NE}}$, it showed a positive moderate correlation around 0.30 for the shortest duration range of [33-44] minutes and the longest duration range of [63-94] minutes. However, for the duration ranges between 45 and 63 minutes, it showed a weak negative correlation with a correlation value of around -0.045. These findings are consistent with the results for the number of queries and visited documents, which suggest that the named entity variation of the framework does not provide reliable estimations of the knowledge gain and its performance does not show any correlation with the session duration.

4.7 Discussion

Our analysis of different implementations of RULK for estimating knowledge gain during a search session reveals that the keyword-based and language model-based approaches, represented by $\texttt{RULK}_{\texttt{KW}}$ and $\texttt{RULK}_{\texttt{LM}}$ respectively, show a moderate positive correlation with the real user's knowledge gain, indicating their accuracy in estimating knowledge gain. They reported a correlation of around 0.3, with the $\texttt{RULK}_{\texttt{LM}}$ slightly lower than $\texttt{RULK}_{\texttt{KW}}$. Given that the language model is designed to capture semantic relationships between words, we expected it to have a higher performance, when compared to vocabulary keyword representation. One possible interpretation of these results is that our keyword-based model captured only the ten most important keywords in the target document and evaluated the user's knowledge accordingly. Our language model on the other hand captures all the information in the target document and compares the user's knowledge accordingly.

On the other hand, the named entity-based approach, represented by $\texttt{RULK}_{\texttt{NE}}$, shows a very weak positive correlation of around 0.1, suggesting that it can not be solely reliable in estimating the user's knowledge gain compared to the other two models. This could be due to the focus of the named entity-based approach solely on capturing specific entities mentioned in the text, which may not always be reflective of the subject topic.

When combining different approaches, we observe that there is an improvement in the accuracy of the estimations. The combination of the three approaches as proposed in Equation 4.9, results in the highest correlation values, indicating that capturing the strengths of different models can elevate the representation of the knowledge. While the named entity-based approach alone did not perform well, it is evident that it adds complementary information to the other approaches, as seen in the improved correlation values when NE is included in the combinations. Although the contribution of theNE coefficient $(1-\alpha-\beta)$ in the mixed model was relatively small, with a value of around 0.16, adding it to the combined model provided an additional benefit. It is also worth mentioning that the difference between keywords and entities lies in their level of specificity, with entities being more precise as they designate specific concepts. This further highlights the potential benefits of incorporating different representations, including entities from a knowledge graph, in estimating knowledge gain accurately during a search session.

The results suggest that the proposed representation of knowledge gain through multiple representations, including keywords, language models, and named entities, can capture different

aspects of the text and enrich the estimation, leading to improved performance in estimating the user's knowledge gain.

It is also worth mentioning that the difference between keywords and entities lies in their level of specificity, with entities being more precise as they designate specific concepts. This further highlights the potential benefits of incorporating different representations, including entities from a knowledge graph, in estimating knowledge gain accurately during a search session.

As for the impact of the session duration on the accuracy of the estimations, the results of the correlation analysis for session length in terms of the number of queries, number of visited documents, and duration of the session do not show any consistent pattern for all three implementations, namely $\texttt{RULK}_{\texttt{KW}}$, $\texttt{RULK}_{\texttt{LM}}$, and $\texttt{RULK}_{\texttt{NE}}$. The lack of a clear pattern in correlation values for session length across all three implementations suggests that session length does not significantly impact the effectiveness of $\texttt{RULK}_{\texttt{KW}}$ and $\texttt{RULK}_{\texttt{LM}}$ in estimating knowledge gain. On the other hand, $\texttt{RULK}_{\texttt{NE}}$ shows inconsistent correlation values ranging from negative weak correlations to moderate positive correlations without exhibiting any clear trend. This result is not surprising, considering the findings from [RQ4-1] and [RQ4-2], which showed that named entities alone may not be sufficient to capture the user's knowledge and estimate their knowledge gain accurately.

The investigation into the effect of session length on the performance of the RULK framework was needed, as the representations of the user's knowledge become denser with longer sessions. The count vectors in the vocabulary learning and the named entity become higher as the count of the keyword read increases. In the case of language models, the representation gets updated by adding the language vectors, which becomes heavier. Our results indicate that the performance of the framework is not affected by the number of queries, visited documents, and session duration. Further research could explore even longer sessions where the search task is accomplished across multiple sessions while considering the decay factor.

The Table 4.2 displays the correlation between the real gain from one side and RPL and ALG from the other side. To assess the significance, we conducted an ANOVA test, which resulted a p-value of 0.835. This result suggests that there is no significant difference at the $p < .05$ level. Therefore, we can infer that both RPL and ALG effectively capture similar aspects of real knowledge gain.

The RULK framework supports capturing the user's current knowledge state and is capable of representing their previous knowledge, if known. While previous research has demonstrated the importance of accounting for the user's previous knowledge when considering their knowledge state, one potential area for future work is to investigate the significance of the user's previous knowledge in the estimations made by the RULK framework. In our study, where users were assigned topics they knew the least about, we assumed that the user's previous knowledge would have minimal influence. However, it would be intriguing to evaluate the RULK framework using a dataset where the user's previous existing knowledge is significant.

4.8 Conclusion

In this chapter, we have proposed RULK, a framework for **R**epresenting **U**ser **L**earning and **K**nowledge, which captures the knowledge acquired by users through the documents they read during search sessions. The representation of user knowledge is dynamic and is updated as new

information is obtained from reading new documents. Additionally, the framework includes the ability to estimate the user's knowledge gain with respect to a specific learning objective.

Those capabilities are handled by the three main components of RULK : The Feature Extractor γ generates embeddings from the documents clicked by users, the Updater σ maintains a vector representing the user's knowledge, and the Estimator θ estimates the amount of knowledge gained by the user during their search session.

The user's representation received the most attention in this framework. This task has not been extensively explored in the literature previously, but is important as it is later utilized in retrieval algorithms to determine the relevance of documents and whether they should be presented to a user. To represent the user's knowledge, we have explored various methods based on the knowledge they acquire from the documents they visit during their search sessions. We implemented three variants : RULK_{KW}, RULK_{LM}, and RULK_{NE}, each relying on a different method for knowledge representation. They respectively use extracted keywords, transformer-based language models, and named entities.

To update the user's knowledge, the content of a visited document is added to the representation of the tracked user's knowledge. To estimate the amount of knowledge gained by the user in a search session, with a specific focus on a goal-oriented topic, we compare the similarity between the user's knowledge and the representation of a reference target document that is assumed to contain all the relevant information about the topic.

To evaluate the effectiveness of our proposed framework, we conducted experiments on a dataset of 500 users who were using the Web to learn about specific subjects. Our results showed that RULK_{KW} and RULK_{LM} implementations can, to a certain degree, accurately estimate the real user knowledge gain. As for RULK_{NE}, representing the user's knowledge solely with named entities was found to be insufficient for accurately estimating the knowledge gains. However, by combining the named entities approach with the other approaches, the results of the latter could be improved by 10%. We proved then that named entities' representations are complementary to the other representations. Our results also showed that the accuracy of the framework is not affected by the session length.

CHAPTER 5

Study II : Developing a User-Centric Evaluation Measure Based on Learning Objectives

The current evaluation measures for information retrieval algorithms do not fully consider the cognitive aspects and dynamics of users, as they often consider an isolated query-document environment. To evaluate retrieval algorithms that help users in learning, there is a need for a measure that can assess the potential learning outcomes after reviewing the search results. This implies that the measure should take into account the user's knowledge state at the beginning of the search session and the knowledge state they aim to achieve by the end of the session. The measure should also be aware of the knowledge changes that are occurring during the search session to evaluate the subsequent results accordingly. However, existing relevance measures often focus on a single search session, query, or goal. In this chapter, we propose a novel evaluation measure that considers the user's evolving knowledge state and learning goals during information retrieval. Our approach uses a vocabulary learning model that tracks the user's knowledge states using the occurrences of topic-related words that the user reads in the search result. To evaluate the relevance of a document, its content is compared to the user's knowledge state at the time of the search, as well as to the desired knowledge state. The relevance of a document varies among users because they start the search with different background knowledge about the topic. We demonstrate the effectiveness of this measure by comparing it to the knowledge gain reported by 500 crowd-sourced users who searched the web across 10 different topics. We also explore how considering the user's prior knowledge affects the accuracy of our measure.

5.1 Introduction

The search as learning field has increasingly acknowledged the significance of incorporating the user's knowledge into the search process. Rose *et al.* highlighted the necessity of adapting current algorithms and interfaces to consider users' knowledge and incorporating these elements into ranking algorithms (Rose & Levinson, 2004b). This remains a significant challenge to this day (Culpepper, Diaz, & Smucker, 2018), as they can greatly influence learning behavior during information retrieval. This user's knowledge state can be reflected by several factors such as the number of submitted queries (Sanchiz, Chin, et al., 2017), query length (White et al., 2009), and vocabulary used (Vakkari et al., 2003). Additionally, Roy *et al.* found that a user's prior knowledge of the searched topic can impact their learning curve during the search sessions (Roy et al., 2020). Therefore, we advocate for the integration of not only the user's knowledge acquired during search sessions but also their prior knowledge at the beginning of the search, into retrieval and evaluation measures.

While some work has incorporated the user's knowledge into the profile representation or retrieval algorithm, little attention has been given to incorporating it into evaluation measures. For example, Syed and Collins-Thompson tracked the user's knowledge to optimize retrieval algorithms with the aim of maximizing learning gain while minimizing the effort spent on reading documents (Syed & Collins-Thompson, 2017a). Also, in our previous work (El Zein & da Costa Pereira, 2020a), we proposed a search retrieval filter that aims to strike a balance between returning documents that are novel to the user's knowledge, but not excessively so. The mentioned filter tracks the user's knowledge by considering a set of weighted keywords.

Traditional evaluation measures such as "precision" and "recall" are inadequate for capturing the relevance of documents for learning outcomes, as they assume a common relevance for all users regardless of their background knowledge and information needs. Additionally, existing contextual retrieval measures lack a standardized evaluation metric and fail to consider the dynamic changes in the search context after the user has interacted with a document (Borlund, 2003b). Previous studies such as (Syed & Collins-Thompson, 2017b) have employed user-based approaches to evaluate retrieval algorithms through experiments conducted in various setups and comparing the learning outcomes of users at the end of the session. User-based evaluation approaches can be time-consuming, expensive in terms of resources, and difficult to replicate, especially when users have different knowledge states. Batch laboratory-based evaluations cannot be used in this context as they rely on relevance judgment files containing pairs of documents and their respective relevance judgments. However, in search as learning, there is no unique relevance for a document as it might depend from one user to another. A relevance judgment file might be convenient to evaluate SAL algorithms if it contains a triple comprising the user, the document, and the relevance of the document for that specific user. Unfortunately, such resources are not readily available.

In this chapter, we present a revised definition of document relevance inspired by Goffman *et al.*'s emphasis on the relationship between the relevance of a document with the documents that preceded it (Goffman, 1964). The proposed definition evaluates documents with respect to their potential learning outcome with respect to a user with a specific knowledge state and need. Our proposed definition is also aligned with Boyce *et al.*'s statement that the most relevant document should not only be topical but also novel (Boyce, 1982). According to these authors, the impact of a document on the user's knowledge state should be reflected in the selection of documents for subsequent positions. We extend this measure to evaluate ranked lists too, taking into account the dynamic nature of the user's knowledge during the session. We use a vocabulary learner model that

counts the occurrences of topic-related words the user reads in the search results. The relevance of a document is performed by evaluating the content of the document to user's model and need. Additionally, the proposed evaluation monitors their progress towards reaching their learning goal by tracking the occurrences of the words that they should ideally see based on their learning objectives.

We also discuss in this chapter, the ranking evaluation method proposed by Clarke *et al.* (Clarke et al., 2008) for assessing novelty and diversity in ranked search results. This method evaluates the ranked list by considering the information presented in previous documents, as well as the information understood by users. We acknowledge the significance of this contribution and draw inspiration from it in our own work. However, we aim to extend the limitations of this approach, as it treats documents as sets of information nuggets that may or may not be present, without considering the potential repetition of information across documents. Our proposed measure takes this into account and addresses this limitation. Additionally, all the previous methods often limit relevance assessment to a single search session, query, or search goal, without considering the user's ability to submit different queries and update their knowledge, which can impact subsequent queries. Our measure accounts for these limitations and provides a more comprehensive evaluation of novelty and diversity in ranked search results.

This chapter aims to answer the following research questions :

RQ5-1 How effective is the proposed measure in evaluating the learning gain resulting from reading a ranked list of search results ?

RQ5-2 How does incorporating the user's knowledge state at the beginning of a search session affect the effectiveness of the proposed measure ?

To address these research questions, we evaluate the proposed measure using a real-world search as learning dataset consisting of the logs of 500 users (Gadiraju et al., 2018) who used the Web to acquire knowledge on ten different topics and information needs. The user's knowledge states were quantified by calibrating scientifically formulated knowledge tests taken before and after search sessions. We adapted the dataset to fit our study.

5.2 Related work : evaluating ranked lists by rewarding novelty and diversity

In this section, we discuss the evaluation measure proposed by Clarke *et al.* (Clarke et al., 2008). The presented work argues that evaluation measures should align with user requirements, especially when tuning IR systems and learning ranking functions. The authors of the mentioned work assert that current evaluation measures do not accurately deal with ambiguity in queries and redundancy in retrieved documents. To address these issues, the authors develop a specific evaluation measure based on cumulative gain.

The presented measure evaluates ranked search results by considering the information in the document as nuggets $u \subseteq N$, where $N = \{n_1, n_2, \ldots, n_m\}$ is the space of possible nuggets. A nugget is considered as a binary property of a document : a document either contains the information or not.

The user's information need was similarly represented as a set of nuggets $d \subseteq N$. A document was considered relevant if it contained at least one nugget that is also contained in the user's information need. Furthermore, an assumption was made that a user's interest in one nugget is independent of other nuggets. The conditional probability of a document d to be relevant ($R = 1$),

given that the nugget n_i belongs to both sets u and d, is noted as follows :

$$P(R = 1|u, d) = P\left(\exists n_i \text{ such that } n_i \in u \cap d\right) \tag{5.1}$$

The presented work evaluated the gain of a document at a rank k with respect to the content of the document and the user's information need. In order to reward novelty and penalize redundancy in a ranked list, it was assumed that if a specific nugget appears in the first $k-1$ documents, then a repetition in d_k will provide no additional benefit : redundancy should be avoided in favor of novelty. The gain vector G of the k^{th} document is defined as follows :

$$G[k] = \cdot \sum_{i=1}^{m} J(d_k, i) \cdot (1-\alpha)^{r_{i,k-1}} \tag{5.2}$$

where $J(d_k, i)$ is the binary judgment done manually by an assessor whether the nugget i is in the document d at position k. It is equal to 1 if the assessor has judged that d contains nugget n_i, and $J(d, i) = 0$ if not. The value α is a constant with $0 < \alpha \leq 1$, reflecting the possibility of an assessor error in their judgment. This definition assumes that positive judgments may be erroneous, but that negative judgments are always correct.

Clarke *et al.* employed the classic cumulative gain CG and discounted cumulative gain metrics DCG to assess the relevance of a ranked list with regards to novelty and diversity. In this chapter, we drew inspiration from this previous work, which made notable strides in considering both user needs and document content when assessing information retrieval. Nonetheless, we identified some limitations in this approach and sought to develop an evaluation measure that could overcome them. Clarke *et al.* employed the classic cumulative gain CG and discounted cumulative gain metrics DCG to assess the relevance of a ranked list with regards to novelty and diversity. After we propose our own relevance and gain measure, we will follow Clarke *et al.*'s approach by utilizing cumulative and discounted cumulative gains to evaluate ranked lists. These methods have been widely employed as evaluation measures in information retrieval and offer several advantages. Specifically, DCG effectively accounts for the relevance of items in a ranked list by assigning higher scores to more relevant items, emphasizing the importance of placing highly relevant items at the top. Additionally, DCG incorporates the position of each item in the ranking, reflecting the decreasing significance of items as we move down the list. This characteristic captures the importance of the ranking order and penalizes systems that prioritize less relevant items at higher ranks. Furthermore, DCG is considered as a flexible measure that is easy compare with different systems.

These limitations first included a narrow view of relevance by considering a document relevant only if it contained a piece of information that was not previously proposed to the user. While this approach favored novelty, it did not consider the usefulness of repeating information for learning. Another limitation was that the measure did not take into account the user's previous knowledge. It only measured the relevance of a document based on the information it contained without considering what the user already knew about the topic. As a result, the relevance of a document could be the same for all users, even if they had different levels of knowledge about the topic. Finally, the evaluation measure only considered the relevance of documents within a single session and did not take into account what had been previously proposed to the user in previous sessions. This fails to capture the user's long-term learning needs and could miss proposing some real novel content to the user.

5.3 Proposed measure : evaluating ranked lists for users with learning objectives

In this section, we introduce our proposed definition for relevance and the related evaluation measure that relies on the user vocabulary learner model discussed earlier in Section 4.4.1. We establish the foundation of our work by defining our notion of relevance :

> We refer to document relevance as the degree to which it can help a user gain knowledge or make progress toward their learning objective. It is a three-part subjective measure that encompasses : the knowledge in a document, the user's current knowledge state, and their intended knowledge state. Unlike the traditional definition of query-document relevance, there is no universal relevance of a document to all users, as it is relative to the user's individual knowledge and goals.

To better illustrate our approach, we provide a motivating example along with relevant illustrations.

5.3.1 User knowledge model

In our framework, we adopt the vocabulary learning model, discussed in Section 4.4.1. In this model, the user's learning objective is to learn about a subject T and its related keywords K_T. The vector $\overrightarrow{tks}$, which represents the *target knowledge state*, denotes the frequency of keywords that a user needs to encounter to achieve mastery of the topic. The user's *current knowledge state* is represented by a vocabulary count vector, denoted as $\overrightarrow{cks}$, which keeps track of the occurrence count of keywords related to the topic that the user has read. Each time the user encounters a word kw_i from the topic, the count in cks_i increases monotonically. We make a simplifying assumption that no memory loss takes place during the search sessions.

In this chapter, we present the concept of the "**n**eeded **k**nowledge **s**tate" (nks), which refers to the personalized target knowledge state denoted as $\overrightarrow{nks} = \{nks_1, nks_2, \ldots, nks_m\}$. It is calculated by subtracting the user's current knowledge state $\overrightarrow{cks}$ from the target knowledge state $\overrightarrow{tks}$. Essentially, the variable nks_i represents the **frequency** with which the user *still* has to read the i-th keyword of K_T to reach the target tks_i of $\overrightarrow{t_{ks}}$.

By considering $\overrightarrow{nks}$ as a personalized target state, we acknowledge that users' information needs and search goals are influenced by their prior knowledge and context. Users with different levels of prior knowledge may require varying amounts of new information to reach their desired knowledge state. Therefore, personalized target states that take into account individual differences, such as $\overrightarrow{nks}$, can provide a user-centric method to assess the effectiveness of retrieval algorithms in facilitating users' learning. The needed knowledge state is defined as follows :

$$\overrightarrow{nks} = \overrightarrow{tks} - \overrightarrow{cks} \tag{5.3}$$

It should be noted that all three vectors in the above equation, along with the document vector $\vec{v_d}$, are embedded in the same space and have a fixed length denoted as m. Recall that a document d is represented as a vector $\overrightarrow{v_d} = \{dko_1, dko_2, \ldots, dko_m\}$ where dko_i denotes the the number of occurrences of the i-th keyword kw_i in the document.

5.3.2 Motivating example

To illustrate our approach, we consider a use case in the context of the Web. We have two users, User A and User B, who share the same search goal to learn about the subject of $T = Blockchain$. This goal is represented by the following set of vocabulary : $K_T = \{Cryptocurrency, Bitcoin, Peer - to - peer, Authentication, Encryption\}$ with the following target knowledge state $\overrightarrow{t_{ks}} = \{20, 17, 13, 13, 9\}$. The target knowledge state shows the desired number of occurrences for each of the keywords kw_i in an ideal retrieved document. We chose this set based on the closest semantic associations with the term "Blockchain", identified using a word embedding model trained on English Wikipedia. The semantic similarity threshold is set to 0.6 to keep a reasonable compromise between two competing goals. On one hand, we want to select a set of vocabulary terms that are closely related and useful to the topic of "Blockchain". On the other hand, we do not want to choose terms that are very similar to the topic that they essentially duplicate it, as this would limit the diversity of the retrieved keywords. To keep the example simple, we manually selected the vocabulary size m = 5 .

Table 5.1 presents the top ten Web pages retrieved by a leading commercial search engine in early 2022 for the Web query *"blockchain"*. The table displays the number of occurrences of each keyword kw_i for each Web page. It is worth noting that keywords with similar origins but varying tenses or forms are grouped together, such as "bitcoin" and "bitcoins", "encrypted" and "encryption". The order of the displayed results in the table corresponds to the ranking provided by the search engine.

Document Title	**Cryptocurrency**	**Bitcoin**	**Peer-to-peer**	**Authentication**	**Encryption**
a. Blockchain.com - The Most Trusted Crypto Company	0	2	0	0	0
b. Blockchain Definition : What You Need to Know - Investopedia	4	5	0	0	0
c. Blockchain - Wikipedia	32	34	8	1	0
d. What is blockchain ? | Euromoney Learning	0	4	0	1	0
e. What Is Blockchain Technology ? How Does It Work ? | Built In	20	24	1	0	1
f. Enterprise Blockchain Solutions & Services | IBM	0	0	0	0	1
g. Making sense of bitcoin, cryptocurrency and blockchain - PwC	10	4	1	0	0
h. Blockchain explained... in under 100 words - Deloitte	1	6	0	0	0
i. What Is Blockchain Technology ? A Step-by-Step Guide	34	67	8	0	0
j. The Truth About Blockchain - Harvard Business Review	3	19	2	0	0
Vocabulary knowledge need $\overrightarrow{nks_A}$	**20**	**17**	**13**	**10**	**4**

TABLE 5.1 – The top ten documents returned for the query "blockchain" with keyword occurrences dko and the needed knowledge state for user A.

Figure 5.1 shows the initial knowledge state, $\overrightarrow{c_{ks}}$ of users A and B before the start of a search session. Despite sharing the same search goal, each user has their own distinct previous knowledge. This previous knowledge influences the number of occurrences they need to read in order to achieve their goal. For instance, User A has prior knowledge of the keyword *authentication* with 3 instances and needs to read 10 more occurrences to achieve the target knowledge state. On the other hand, User B has no prior knowledge of this keyword and has to read 13 occurrences of it to achieve the same target knowledge state.

Suppose that we focus solely on the keyword "cryptocurrency". At the start of the search session, user A's need for the word "cryptocurrency" is partially satisfied by document (g) in Table 5.1, which contains 10 occurrences out of a total need of 20. In contrast, document (g) exactly fulfills user B's need for the same word with 10 occurrences. This implies that document g in Table 5.1 is relatively more relevant to user B than to user A. The relevance of the document

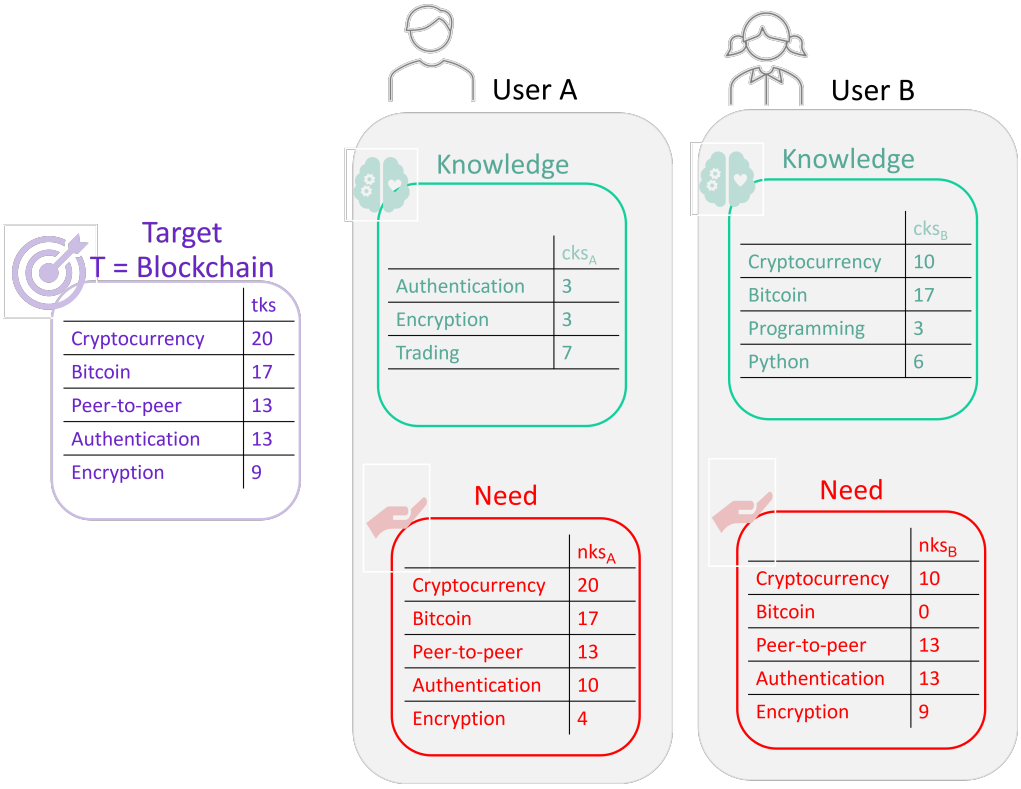

Figure 5.1 – Example illustration showing two users with the same learning goal and different background knowledge.

cannot be determined in absolute terms as it depends on the knowledge and preferences of the respective users.

The personalized vector representing the vocabulary occurrence needs for user A on the topic of "Blockchain" is denoted as $\overrightarrow{nks}_A$. An algorithm that is considered *ideal* should assist user A in achieving their target knowledge state $\overrightarrow{tks}$ in an efficient manner, while also preventing redundancy in information that has already been obtained from previously read documents. As evident from Table 5.1, the search engine initially ranked document (a) as the top result. However, upon comparing the keyword occurrences in the document $\overrightarrow{v_d}$ with the needed occurrences $\overrightarrow{nks}_A$, it becomes intuitive that document (a) is not the most relevant document to fulfill the user's information needs. It only provides 2 occurrences of the keyword "Bitcoin" compared to the user's requirement of 17, and it does not contribute to learning other needed keywords. Documents (c) and (e) could potentially have higher relevance since they provide occurrences that are closer to the user's needs.

User A will be our focus for the rest of this chapter.

5.3.3 Gain brought by one document

We define $Gain(\overrightarrow{v_d}, \overrightarrow{nks})$, a value between 0 and 1, that measures how much a user with a knowledge need $\overrightarrow{nks}$ can learn from a document d. This also can be seen as a relevance measure for the document d for a specific user. Let $gain(kw_i, d)$ denote the learning gain brought by a keyword kw_i in that document to that user.

We first compare the number of occurrences dko_i of each term kw_i to the nks_i in the needed knowledge vector, and use the proportion $\frac{dko_i}{nks_i}$ for this comparison. We define three cases :

— Case 1 : $dko_i = nks_i$. If the document contains the exact number of occurrences needed for kw_i, the gain would ideally be 1.

— Case 2 : $nks_i = 0$. If the user has acquired all the necessary knowledge about the keyword, there is no additional advantage gained from decreasing the number of times the keyword occurs. The gain should be 0.

— Case 3 : $dko_i < nks_i$. If the document provides fewer occurrences of a keyword than needed. The ratio would be in this case smaller than one. In this case, the learning gain should continue to increase until the ratio becomes equal to one, indicating that the desired number of occurrences has been reached.

— Case 4 : $dko_i > nks_i$. If the document provides more occurrences of a keyword than needed, the gain will still be 1. In this case, the learning gain may start to decrease as the excess occurrences of the keyword become redundant and less beneficial for the user's learning. This is known as the *law of diminishing returns*, where additional input leads to a decreasing output (in this case, learning gain). This approach also takes into account the effort expended by the user, as proposed by Yilmaz *et al.* (E. Yilmaz, Verma, Craswell, Radlinski, & Bailey, 2014).

To translate those cases, the related formula represents a continuous curve showing an upward-sloping pattern gradually approaching one as $\frac{dko_i}{nks_i}$ approaches one, and then starts to decline and flatten when $\frac{dko_i}{nks_i}$ is greater than one. This is because when $\frac{dko_i}{nks_i}$ is less than one, the learning gain increases rapidly as the number of occurrences of the keyword increases. However, as $\frac{dko_i}{nks_i}$ approaches one, the increase in learning gain becomes less significant, and when $\frac{dko_i}{nks_i}$ is greater than one, the learning gain may start to decline due to the law of diminishing returns. Therefore,

the formula should account for all four cases mentioned previously, using the appropriate function to represent the learning gain in each case.

This ensures that redundancy is avoided and novelty is promoted in the evaluation of information provided by a document. This formula encourages higher gain values when the occurrences of the keyword in the document are closer to fulfilling the user's need, but penalizes excessively high occurrences, resulting in diminishing returns. A document is considered relevant or useful if it contains at least one keyword that can help decrease the user's occurrence need.

We propose the following equation and present it in Figure 5.2 for visualization :

$$gain(kw_i, d, nks_i) = \begin{cases} \frac{dko_i}{nks_i} e^{1-\frac{dko_i}{nks_i}}, & \text{if } nks_i > 0; \\ 0, & \text{if } nks_i = 0. \end{cases} \tag{5.4}$$

The overall gain, with respect to all the document's keywords, is then given by the following equation :

$$Gain(\overrightarrow{v_d}, \overrightarrow{cks}) = \sum_{i=1}^{m} gain(kw_i, d, nks_i). \tag{5.5}$$

For our example, the $Gain(\overrightarrow{v_d}, \overrightarrow{cks_A})$ of each of the 10 documents to the user A is respectively $0.28, 1.04, 2.76, 0.75, 2.66, 0.53, 1.52, 0.80, 1.96$ and 1.70.

By calculating the gain of each document using Equation 5.5, we can determine the most "relevant" document or the one that should be returned first. In the above example, document (c) has the highest individual gain and should be returned first. The ordered list of documents by their individual gain would be : $c - e - i - j - g - b - h - d - f - a$.

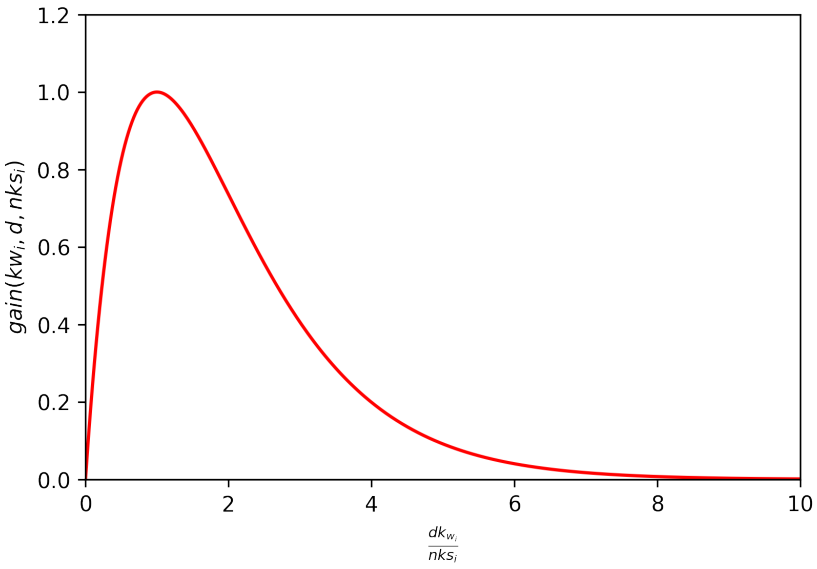

Figure 5.2 – A drawing of the function $gain(kw_i, d, nks_i)$ - Equation 5.4.

This formula promotes novelty and avoids redundancy in evaluating information from a document. It encourages higher learning gains when the document provides a higher number of occurrences of the keyword that is closer to fulfilling the user's need, while discouraging exces-

sively high occurrences, which results in diminishing returns. A document is deemed relevant or useful if it contains at least one keyword that can assist in reducing the user's occurrence need.

5.3.4 Gain brought by a document at rank r

The previous section only considered the gain of a single document independently, without considering the gain obtained from previously read documents. The proposed gain formula, Equation 5.5, allows determining the document that should be ranked first. For subsequent documents, such as the second document and onwards, it is necessary to recalculate the user's information need and update the knowledge state based on the information gained from the previously read document (at rank $r-1$), before calculating the gains again. This ensures that the user's evolving knowledge state is taken into consideration when determining the gain of each document and ranking them accordingly. To update the $i-th$ element of the user need at rank r, $nks_i[r]$, we propose the following equation :

$$nks_i[r] = \begin{cases} nks_i[r-1] - dko_i & if\ nks_i[r-1] > dko_i \\ 0 & if\ nks_i[r-1] \leq dko_i \end{cases} \tag{5.6}$$

The intuition behind this equation is that the count of the user's need nks_i for the keyword kw_i will monotonically decrease for every occurrence of it dko_i in the document. If the need for the keyword is less than or equal to the number of occurrences in the document, then the user's need is satisfied and becomes equal to zero.

After the $\overrightarrow{nks}[r]$ got updated thanks Equation 5.6, the gain of the document at rank r can be calculated. We can define the gain at rank r, $Gain[r]$, as the gain provided by the document :

$$Gain[r] = Gain(\overrightarrow{v_d}, \overrightarrow{nks}[r]) \tag{5.7}$$

The documents with a higher gain will have a higher ranking, and this process continues until all documents are ranked.

For example, the knowledge state of user A, after reading document (a) at rank $r = 1$, gets updated by adding all the keywords of document (a) and their occurrences. Applying Equation 5.6 will update the values for $\overrightarrow{nks}_A[2] = \{20, 15, 13, 10, 4\}$. The learning gain brought by the next document (i.e. document (b)), at rank r = 2, is $Gain[2] = 1.09$. We notice that the gain provided by the document (b) after reading (a) has now a higher effect than reading the document (b) alone because it covers a higher proportion of the user's need now : 5 *bitcoin* occurrences out of the need for 15 compared to 17 if read alone. Therefore, the Gain vector is $\langle$ 0.28, 1.09, 2.19, 0.27, 0.97, 0.64, 0.52, 0, 0.5, 0 $\rangle$. Equation 5.6 updates the knowledge need, which decreases as the user reads more documents, whereas the Gain vector represents the gain provided by each document, which may not necessarily decrease. As Figure 5.2 shows : as long as the proportion is smaller than 1, the higher the proportion the higher the gain. For example, the user's need for *Bitcoin* was 17, document (a) has 2, and document (b) has 5. If the user reads document (b) first, the proportion will be 5/17 = 0.29. If the user read document (a) then (b), the proportion for (b) will be 5/(17-2) = 5/15 = 0.3 > 0.29.

This approach enables a dynamic ranking of documents based on the user's evolving knowledge state, ensuring that the most relevant documents are presented to the user based on their information needs at each stage of the information retrieval process.

5.3.5 Cumulative gain

We calculate the *cumulative gain vector* (Järvelin & Kekäläinen, 2002) at rank r, $CG[r]$, as follows :

$$CG[r] = \sum_{j=1}^{r} Gain[j] \tag{5.8}$$

For our example, we have the following cumulative gains : $CG = \langle$ 0.28, 1.37, 3.56, 3.83, 4.8, 5.44, 5.96, 5.96, 6.46, 6.46 $\rangle$. To account for the rank of the documents, before computing the cumulative gain vector, we apply a discount to penalize documents lower in the ranking. We use the classical discount : $log_2(1 + r)$, and define *discounted cumulative gain* at rank r as follows :

$$DCG[r] = \sum_{j=1}^{r} \frac{Gain[j]}{log_2(1 + j)} \tag{5.9}$$

For our example, we get $DCG = \langle$ 0.93, 2.87, 5.91, 5.48, 6.04, 6.44, 6.60, 6.25, 6.46, 6.20 $\rangle$.

We will now normalize the Discounted Cumulative Gain measure *nDCG*. The ideal ranking is the order that maximizes cumulative gain *CG* at any rank. In our example, the ideal rank is : $c - e - j - g - f - d - a - b - h - i$. The associated ideal gain vector is : $Gain' = \langle$ 0.76, 0.97, 0.93, 1, 0.65, 0.27, 0, 0, 0, 0 $\rangle$, the ideal cumulative gain vector is $CG' = \langle$ 2.76, 3.73, 4.66, 5.66, 6.31, 6.58, 6.58, 6.58, 6.58, 6.58 $\rangle$, and the ideal discounted cumulative gain vector is $DCG' = \langle$ 9.17, 7.82, 7.74, 8.10, 8.11, 7.79, 7.29, 6.90, 6.58, 6.32 $\rangle$. Finally, we normalize the calculated DCG by the ideal discounted cumulative gain vector DCG' :

$$n - DCG = \frac{DCG}{DCG'} \tag{5.10}$$

For our example, $n - DCG = \langle$ 0.10, 0.37, 0.76, 0.68, 0.74, 0.83, 0.91, 0.91, 0.98, 0.98 $\rangle$.

5.4 Dataset II : Knowledge gain in informational search sessions

In this section, we describe the experimental setup utilized for the dataset we adopted. The study performed by Ujwal *et al.* (Gadiraju et al., 2018) addressed the users' knowledge gain in informational search sessions. The study recruited 500 distinct users on a crowd-sourced platform to perform orchestrated real-world search sessions spanning over 10 different topics and information needs. The user's knowledge state was quantified by calibrating scientifically formulated knowledge tests taken before and after search sessions. The study resulted in a public dataset *that we adapted it to suit our experimental requirements.

5.4.1 Participants and study population

The study participants were recruited from the *CrowdFlower* † crowd-sourcing platform. They were required to be located in English-speaking countries and to have an adequate level of proficiency in English, which was verified by *CrowdFlower*. To ensure the resulting data's reliability, participation was restricted to Level-3 workers. CrowdFlower considers Level-3 contributors as the highest-quality workers who have completed over 100 test questions across various types of

∗. https ://sites.google.com/view/knowledge-gain
†. http ://www.crowdflower.com/

tasks and achieved near-perfect overall accuracy. The study involved logging the behavior of 500 users.

5.4.2 Search task and topic assignment

The study participants were given predefined information needs specific to a topic they were assigned and were provided access to a search tool to gather information about it. Participants were encouraged to end the search session only when they believed that their information need had been met and when they were ready to take the post-session test.

These topics were selected randomly from the TREC 2014 Web Track dataset ‡, and the search was conducted against a constructed corpus. The study included 10 proposed topics, denoted as follows : Altitude Sickness (AltS), American Revolutionary War (ARWar), Carpenter Bees (Bees), USS Cole Bombing (Bombing), Evolution, NASA Interplanetary Missions (NASA), Orcas Island (Orcas), Sangre de Cristo Mountains (Sangre), Sun Tzu, and Tornado. Due to space restrictions in some tables, we will use abbreviated names to refer to some topics in the following subsections.

As an example, the information need conveyed for the *Carpenter Bee* topic was :

> In this task, you are required to acquire knowledge about the biological species 'carpenter bees'. How do they look ? How do they live ?

5.4.3 Search tool and logged interactions

The tool used for the study was the *SearchWell* platform, which utilizes the *Bing* Web search API. The search activity of participants, as well as their mouse movements, clicks, and key presses, were logged using PHP/Javascript and the jQuery library.

We extracted, for every user, the set of websites visited during the search session. We refer to a website Uniform Resource Locator (URL) as a document or page. We note that, on average, users performed 2.54 ± 1.77 interactions (one query = one document) and read 2.31 ± 1.48 distinct documents.

5.4.4 Knowledge measurement

Multiple-Choice test : For each topic, a test was created consisting of 10-20 statements that participants had to answer with one of the options : True, False, or I don't know. The test scores were originally on a scale of 1 in the dataset, but to simplify dealing with a large number of decimals, we will multiply it by 100.

Pre-session test : Before starting the Web search, participants were required to take a multiple-choice test, also known as a "calibration test", to assess their prior knowledge of the topic before beginning the Web search. They were instructed not to use the Web to look for answers and were encouraged to provide accurate responses rather than guessing. Participants were also assured that their responses would not affect their payment.

Post-session test : After the search session, participants were given an identical test to the pre-session test to assess their knowledge gain and ability to recall facts, without informing them that it was the same test. They were also informed that their accuracy on the post-session test

‡. http ://www.trec.nist.gov/act_part/tracks/web/web2014.topics.txt

The species of 'carpenter bees' derive their name due to their nesting behavior.
❍True ❍False ❍I don't know

There are more than 500 sub-species of carpenter bees.
❍True ❍False ❍I don't know

All carpenter bees are typically sociable and live in large nests.
❍True ❍False ❍I don't know

Figure 5.3 – Three Sample statements from the tests of the *Bees* topic

could impact their bonus payment.

The score of a user's knowledge test was based on the correctness of their answers. The "I don't know" answer was considered incorrect. The user's *actual* knowledge gain was measured as the difference between their pre-session and post-session tests' scores. Figure 5.3 shows a sample of three test statements from the *Bees* topic.

5.4.5 Data manipulation

5.4.5.1 Data filtering and study population

We filtered the data to exclude users who did not interact with the system during the search session. Specifically, we excluded those who did not click on any search results or did not enter any query, as there was no user-system interaction. We also removed users who demonstrated a negative knowledge gain, to align with our assumption that no memory loss occurred. Ultimately, our filtering process resulted in a final sample of 361 users across all 10 topics. The number of users per topic after data filtering, N, is displayed in Table 5.2. The table also shows the average pre- and post- session test scores, as well as the average knowledge gain per topic.

Topic	**Avg. Pre Score**	**Avg. Post Score**	**Avg. Actual Gain**
AltS (N=39)	55.74 ± 16.23	75.71 ± 13.48	19.97 ± 14.65
American Revolutionary War (N=36)	26.67 ± 22.42	58.61 ± 19.59	31.94 ± 24.0
Bees (N=39)	40.26 ± 28.14	72.56 ± 16.34	32.31 ± 27.57
Bombing (N=36)	21.53 ± 23.35	58.8 ± 14.36	37.27 ± 24.92
Evolution (N=35)	35.0 ± 18.72	53.81 ± 20.35	18.81 ± 13.46
NASA Interplanetary Missions (N=27)	30.93 ± 19.37	56.48 ± 16.97	25.56 ± 18.31
Orcas (N=39)	25.0 ± 28.19	64.62 ± 21.1	39.62 ± 28.98
Sangre de Cristo Mountains (N=35)	24.86 ± 24.18	53.14 ± 18.91	28.29 ± 22.69
SunTzu (N=39)	27.86 ± 25.63	57.09 ± 21.18	29.23 ± 22.97
Tornado (N=36)	27.62 ± 18.9	52.01 ± 18.55	24.38 ± 19.8
Overall (N=361)	**31.55 ± 22.51**	**60.28 ± 18.08**	**28.74 ± 21.74**

TABLE 5.2 – The average user knowledge gain across the different topics. N is the number of users.

5.4.5.2 Internet archive and Web-Scraping

The study resulting in the dataset was conducted on the 25^{th} and 26^{th} of July 2017. To obtain the dataset, we utilized the *Wayback Machine* [§] API to restore a snapshot of each document as it was on the date of the study. If the snapshot of a page was not available at the specified timestamp, the closest snapshot date was used. We manually extracted the content of 38 websites that were not archived by the machine, as of October 2021.

We then extracted the textual content from the websites using the Python library *Beautiful Soup* [¶]. This library is capable of parsing and pulling out data from HTML and XML files. After the extraction, we performed some text pre-processing tasks to exclude non-content sections such as references, related articles, ads, customer reviews, and comments. These sections were typically found at the end of the page. The remaining text included only the main content of the page, as well as image captions, info boxes, and tables of contents. We translated documents written in a language different from English using the *Google Translate* feature.

5.5 Methodology and user study

To assess the effectiveness of our proposed measure, we employ the dataset described in Section 5.4, which includes the logs of the user's interactions with a search engine across 10 different topics, as well as the users' knowledge state before and after the search session, and their learning gain. We aim to verify whether our proposed method effectively assesses the gain in knowledge when compared to the actual learning outcome.

Our validation method has four stages :

1. Extract the *target keyword set* K_T for each of the 10 topics T.
2. Determine the *target knowledge state* $\overrightarrow{tks}$ for every topic. We explore two approaches : the first approach assumes a universal need for all users of the same topic. In contrast, the second approach personalizes this vector by considering the prior knowledge of every user. In the latter approach, each user has their own vocabulary occurrence need vector $\overrightarrow{nks}$.
3. Calculate the discounted cumulative gain using the proposed method in Section 5.3.5, considering the order of visited pages as the rank for evaluation.
4. Compare the gains resulting from our evaluation metric (both personalized and non-personalized) with the user gain reported by the participants.

We will outline the details of our methodology for each of these steps in this section.

5.5.1 Setting the target keyword set K_T and knowledge state $\overrightarrow{t_{ks}}$

In the study of the dataset, the user's objective was to correctly answer test questions related to a specific topic T, related to an explicit information need. Thus, the user needed to acquire the vocabulary terms of the statements who's answer is *True*, in order to answer them correctly answer them. When extracting the target keyword set K_T from test statements, we chose to select *three* words per statement. This decision was based on the length of the statements, which typically average around 15 words, and on the relatively small number of documents visited by users,

§. https ://archive.org/web/
¶. https ://pypi.org/project/beautifulsoup4/

which is typically under 10 unique documents. Selecting three keywords per statement provides users with a fair representation of the question to be answered. Having too many keywords could result in including irrelevant terms. On the other hand, too few keywords may result in redundancy, as different statements discuss the same subject of the topic, potentially resulting in the same keywords being selected for multiple statements. Therefore, selecting three keywords strikes a balance between these two extremes. For this task, we used the *YAKE (Yet Another Keyword Extractor)* method (Campos et al., 2018) that automatically identifies and extracts keywords or key phrases from text documents. The advantage of *YAKE* is that is it designed to work with short texts, such as titles, abstracts, or metadata, and it aims to identify important terms that convey the most relevant information in the given context. After extracting the target keyword set for each topic, we took steps to remove redundant keywords that matched the topic name itself. For example, if the topic was *"NASA Interplanetary Missions"*, we removed keywords such as "NASA", "interplanetary", and "missions". This helped us to refine the final set of keywords, ensuring that they were more specific and relevant to the content of the topic itself, rather than simply repeating its name. This approach also avoided any confusion or ambiguity that might arise if the same word were used both as a topic name and a keyword. Finally, we applied the *Porter Stemmer* algorithm in Python to reduce each keyword to its base or stem form and grouped related words together to avoid duplication in the set. As a result, the size of the target keyword set K_T was $m_T = 3\times$ the number of *True* statements.

We assumed that the number of keywords a user needs to read is equal to the size of the target keyword set, which we denote as m_T. In order to ensure that all the desired frequencies tks_i are given equal importance, we distribute them equally among the keywords.

5.5.2 Setting the user's *needed knowledge state* $\overrightarrow{nks}$

We explore two approaches to define the user's needed knowledge state :

— **Personalized approach** : The personalized approach takes into account the user's prior knowledge and adapts the needed number of keywords for every user $\overrightarrow{nks}$.

— **Non-personalized approach** : The non-personalized approach assumes that all users start with no prior knowledge on the topic and must read the same number of keyword occurrences $\overrightarrow{tks}$.

In order to personalize the user's information needs and account for their previous knowledge, we used the pre-session test scores Given that the dataset only contained pre- and post-assessment scores and not users' answers to individual statements questions, it was not possible to determine the users' prior knowledge of specific statements or keywords. We can assume however that the overall score is reflective of the user's performance on each statement. We use the pre-assessment score to calculate an approximation of the user's knowledge regarding the keywords before the start of the session.

One way we estimate the user's previous knowledge on each keyword is by multiplying the test score with the number of occurrences they should read, as follows :

$$cks_i = \frac{pre\text{-}session\ score}{100} \cdot tks_i \tag{5.11}$$

We also know that $nks_i = \max(0, tks_i - cks_i)$; we conclude therefore the following :

$$nks_i = \max\left(0, \frac{100 - pre\text{-}session\ score}{100} \cdot tks_i\right) \tag{5.12}$$

In the case of a non-personalized approach, the user's previous knowledge is not taken into account, hence tks_i would be equal to nks_i equal to 1.

To extract the keywords from the documents, we first removed *stop-words* using the *NLTK* Python library (Bird et al., 2009). Next, we applied the *Porter Stemmer* from the same library to stem the words, reducing them to their root form. Finally, we used *word count* to extract the keywords and their occurrences, allowing us to obtain the frequency of each keyword in the documents. This pre-processing step helped in obtaining comparable forms of keywords with the ones in the K_T.

5.5.3 Discounted cumulative gain and results comparison

We rely on the user's behavior visiting the documents to construct a list of documents that the user has interacted with during their search session, in chronological order. We use this list of documents as a proxy for the actual list of documents that were returned, and apply our proposed evaluation measure to this list. This approach allows us to evaluate the effectiveness of the ranking algorithm in terms of its ability to provide relevant information to the user based on their behavior during the search session. For each user, we calculated the discounted cumulative gains using two approaches : personalized and non-personalized. The personalized approach took into account the user's previous knowledge on the topic, while the non-personalized approach treated all users' previous knowledge equally.

We evaluated the effectiveness of our proposed evaluation measure, by calculating the DCG and DCG_{np} , and calculating its Pearson's correlation coefficient with the actual user gain. We employed an independent t-test, which is a commonly used statistical method for comparing the means of two groups, to determine the statistical significance of the differences between the two conditions.

5.6 Results

Table 5.3 shows the correlation coefficient between the two approaches DCG and DCG_{np} with the actual gain. The table also indicates the statistical significance of the correlations, denoted as "$p < 0.05$", which implies that the correlations are considered statistically significant. When comparing the correlation between the topics and the actual gain, it is evident that the personalized approach showed a higher correlation with the gain for each topic compared to the non-personalized (np) approach. The correlations of the gains in the personalized approach ranged from moderate to high, with values ranging between 0.332 and 0.734. On the other hand, the correlations in the non-personalized approach were generally low to moderate, ranging from 0.027 to 0.314, except for two topics, namely *ARWar* and *Bombing*, which showed correlations of -0.217 and -0.065, respectively.

The results show that the personalized and non-personalized approaches are effective in capturing the relevance of documents based on the actual gain obtained by users. However, the personalized approach demonstrates a higher correlation with the gain for each topic compared to the non-personalized approach. This suggests that personalization of the user's knowledge needs enhances the effectiveness of capturing document relevance in relation to the actual gain experienced by users. This supports the findings for the research question **[RQ5-1]**, indicating that considering the user's previous knowledge is important for improving the effectiveness of the evaluation framework.

Topic	**(real gain, DCG_{np})**	**(real gain, DCG)**
Altitude Sickness	0.077	0.431
American Revolutionary War	-0.217	0.333
Carpenter Bees	0.027	0.597
USS Cole Bombing	-0.065	0.521
Evolution	0.314	0.332
NASA Interplanetary Missions	0.221	0.366
Orcas Island	0.382	0.734
Sangre de Cristo Mountains	0.065	0.469
SunTzu	0.191	0.484
Tornado	0.115	0.461

TABLE 5.3 – Pearson's correlation coefficient between two DCG approaches and actual gain ($p < 0.05$) for different topics

Topic	**Real gain**	DCG_{np}	DCG	**p-value**
Altitude Sickness	19.97	3.305	1.038	< 0.00001
American Revolutionary War	31.94	4.054	2.600	0.004
Carpenter Bees	32.31	8.813	3.870	< 0.00001
USS Cole Bombing	37.27	6.514	5.347	0.03
Evolution	18.81	4.432	3.001	0.016
NASA Interplanetary Missions	25.56	12.795	10.792	0.106*
Orcas Island	39.62	9.281	7.231	0.047
Sangre de Cristo Mountains	28.29	9.141	7.127	0.009
SunTzu	29.23	10.453	7.747	0.011
Tornado	24.38	8.863	6.780	0.005

TABLE 5.4 – Comparison of real gain and DCG Scores for different topics in personalized and non-personalized approaches with statistical significance analysis (* indicates non-significant differences at at $p< 0.05$)

To further investigate the differences between the personalized and non-personalized approaches, we performed a statistical significance analysis at a confidence level of 0.95. Table 5.4 shows the results, indicating that the difference between the two approaches is statistically significant for all topics, except in the case of the topic *NASA Interplanetary Missions.*

The results of our analysis support the research question **[RQ5-2]**, which questions the impact of the user's previous knowledge on the gain. The statistical significance analysis reinforces the importance of taking into account the dynamic nature of the user's knowledge in evaluating search systems. It further supports the advantage of the personalized approach over the non-personalized approach in capturing the user's cognitive state and providing a more accurate evaluation of system effectiveness. These findings highlight the need to consider the user's prior knowledge as a factor when evaluating the performance of information retrieval systems.

5.7 Discussion and limitations

Our proposed measure differs from classic relevance measures in that it considers a broader definition of relevance. Specifically, we define a three-part subjective measure that includes the knowledge presented in a document, the user's current knowledge state, and their intended knowledge state. In contrast to traditional relevance measures, which focus solely on the relationship between a query and a document, our evaluation framework extends beyond this isolated query-document environment.

Our proposed evaluation measure accounts for the dynamic and fast-changing nature of the user's knowledge state during search sessions. By continuously tracking and updating the user's knowledge state as they read documents, the relevance of subsequent documents is assessed based on what has been previously proposed and the user's current level of knowledge. Our measure utilizes the concept of cumulative gain, which accumulates the knowledge gained from documents and rewards content that contributes to changing the user's knowledge state towards a goal state. Moreover, it penalizes redundant information, particularly when the user's knowledge about that information has reached saturation.

In comparison to the framework proposed by Clarke *et al.* (Clarke et al., 2008), our approach takes into account the user's previous knowledge when the search session starts. During the experimental sessions, the user's score on the pre-session knowledge test was reflective of their level of knowledge. We used this score to personalize the user's need for information, which affected the relevance of the documents and knowledge gain. In practice, existing prediction models in the literature can help estimate the user's knowledge level (X. Zhang et al., 2015 ; R. Yu, Gadiraju, Holtz, et al., 2018b), even if it is not on a granular level. While not accounting for the user's previous knowledge in the measure showed decent accuracy results, including them imprived the measure accuracy by 356%.

Another advantage over Clarke *et al.*'s work is that we acknowledge the need for repetition in learning and acquisition of user knowledge. In our proposed measure, we consider that information in a document or the user's knowledge is not limited to binary nuggets but includes repeated words. However, beyond a certain point, this repetition becomes redundant because the information has already been acquired. This is in contrast to Clarke *et al.*, who considered knowledge as a set of binary nuggets, where a document either contained the nugget or not.

However, there are limitations to our study. Firstly, our study deals with the user's needs and knowledge as weighted sets of keywords, which may be suitable for vocabulary learning tasks but may not fully capture the complexity of knowledge and goals in higher levels of learning. It would have been interesting to test the performance of this measure using other knowledge representations like language models or named entities like the ones proposed by our `RULK` framework in Chapter 4.

Secondly, while the proposed evaluation measure assesses relevance and measures knowledge related to a specific topic and its corresponding keywords, determining the optimal target keyword set size and required frequency, or weights, of occurrence for the user to read can be a challenging task.

Fourthly, Our concept of document relevance is limited to a document being considered "useful" solely based on its contribution to reducing the occurrence of the user's information need. Although the underlying intuition behind this assumption is valid, we recognize the significance of information repetition in reinforcing the user's learning, even if their information need has al-

ready been satisfied, particularly when aiming for long-term learning. It may be worth considering a decay rate or other mechanisms to account for the reinforcement of learning over time.

Finally, it may not be straightforward to compare our evaluation measure with other measures, primarily due to potential differences in the notion of document relevance and the ability to track the user's knowledge and progress toward their goal. Therefore, comparing the actual user's gain with other measures may not be fair.

5.8 Conclusion

We presented in this chapter a novel evaluation measure for retrieval algorithms that accounts for the user's search goal and knowledge. This measure utilizes vocabulary learning model that tracks the frequency of the topic-related keywords read. Our framework tracks the user's knowledge and progress towards their goal during the user-system interaction.

We consider a document relevant if it contains at least one keyword that contributes to decreasing the user's need for a keyword. We measure the gain brought by a keyword in a document using Equation 5.4, which leads to the overall document gain measured by Equation 5.5. Our measure penalizes redundancy of information not only in relation to previously proposed documents during a session, but also with respect to the user's existing knowledge. In doing so, our framework evaluates documents and ranking algorithms based on the user's knowledge and information need, providing a cognitive assessment of the ranking list of documents returned by an information retrieval system.

We conducted a crowd-sourced user study that demonstrated a correlation between our proposed measure's outcome and the real reported gain by the users. Additionally, we showed the significance of accounting for the user's previous knowledge when defining their information need. Our proposed measure represents a first step towards a cognitive assessment of the ranking list of documents in an IR system.

CHAPTER 6

A Benchmark and Baseline for Search as Learning Evaluation

Evaluating retrieval algorithms in the search as learning field can be a challenging task due to the scarcity of available datasets and relevance judgments. To effectively assess these algorithms in aiding user learning, it is necessary to have a benchmark dataset that makes it possible to measure the "relevance" of every proposed document at any given point in a search session. However, most experiments in the search as learning field measure the user's knowledge before and after a search session, providing a snapshot of knowledge states at only two points in time. The existing datasets provide a measurement of the knowledge gained after the user has seen several documents, but it is difficult to measure the contribution of each visited document to this gain. These datasets might allow comparing the effectiveness of retrieval algorithms only at the end of a search session, but not during it. To overcome this limitation, we have developed a benchmark dataset that allows for measuring the relevance of a set of documents with respect to a specific information need. Such a benchmark provides a more details insight into the impact of retrieval algorithms on user learning,, rather than just assessing the total knowledge gained at the end. Each document will be associated with an estimated gain, which allows tracking of users' knowledge evolution on a document-by-document basis. Our goal is for this benchmark dataset to serve as a standardized method for evaluating the efficacy of retrieval algorithms in enhancing user learning. The changes observed when users employ new retrieval algorithms can be compared to the changes recorded in this dataset.

6.1 Introduction and motivation

Comparing the effectiveness of two retrieval algorithms in search as learning requires a comprehensive evaluation strategy. One intuitive approach is to compare the user's learning outcomes when using the two different algorithms. However, the majority of search as learning experiments measure the user's knowledge before and after the search session, providing only two points of measurement (X. Zhang et al., 2011 ; Gadiraju et al., 2018 ; Câmara, Maxwell, & Hauff, 2022a). As a result, comparing the effectiveness of an algorithm requires their results to be compared at the end of the search session, after the user has reviewed a set of documents.

Ideally, we would like to be able to measure the effectiveness of a retrieval algorithm at any point during the search process to understand how it contributes to the user's learning process. Unfortunately, this is not possible because we do not have data about the user's knowledge on a document-by-document level. This lack of data and benchmarks affects the ability to measure the effectiveness of retrieval algorithms at specific points in time. It is essential to understand how retrieval algorithms contribute to the user's learning outcomes in comparison to the baseline. Comparing the effectiveness of retrieval algorithms to the baseline is an essential part of understanding their contribution to the user's learning process.

A frequently used method to evaluate retrieval algorithms is laboratory-based experiments, where a predefined set of documents is provided with relevance judgments assigned by experts. However, the judgments of experts do not represent the relevance of a document for all users. Also, the related judgments are static and fail to capture changes in the user's information needs or understanding as they interact with different documents. Consequently, this approach cannot be applied to evaluate search as learning algorithms due to the absence of suitable relevance judgments.

This chapter tackles the following research question :

— **[RQ6]** How can we develop a benchmark dataset to compare search as learning retrieval algorithms, given the limited availability of resources ?

In this chapter, we propose the development of a benchmark dataset that includes user logs capturing the evolution of the users' knowledge during the session on a document-by-document level. We describe the design and construction of this dataset, including the data collection process and the annotation of contextual information about users' knowledge levels. The proposed dataset includes measures of document relevance based on the gain they contribute to the user's knowledge. This dataset is intended to serve as a benchmark for evaluating the performance of retrieval systems in facilitating user learning. The proposed resource can contribute to understanding how retrieval algorithms support users in their learning journeys.

6.2 Dataset resource and preparation

To construct the benchmark presented in this chapter, we utilized the dataset introduced in Section 5.4, which was originally proposed by Gadiraju *et al.* (Gadiraju et al., 2018). This dataset contains log information from 500 users who used a search tool to achieve specific learning goals. It provides insight into users' learning behavior, including the documents they visited and the queries they issued, as well as measurements of their knowledge before and after the search session with respect to 10 different topics. The user's knowledge before and after the search session was measured using multiple-choice questions containing 10-20 statements about the topic. The statements were in a *True-False-I don't know* format, and the users had to answer them. As a result, we

Query	urlID
charles darwin	d54, d55
evrim teorisi	d38,d39
modern synthesis	d59,d60
teoría de la evolución	d6,d9,d17,d51,d52
theory of evolution	d1,d2,d3,d4,d10,d15,d18,d22,d57
thomas henry huxley	d12

TABLE 6.1 – Sample of the ARWar\querydoc _index file.

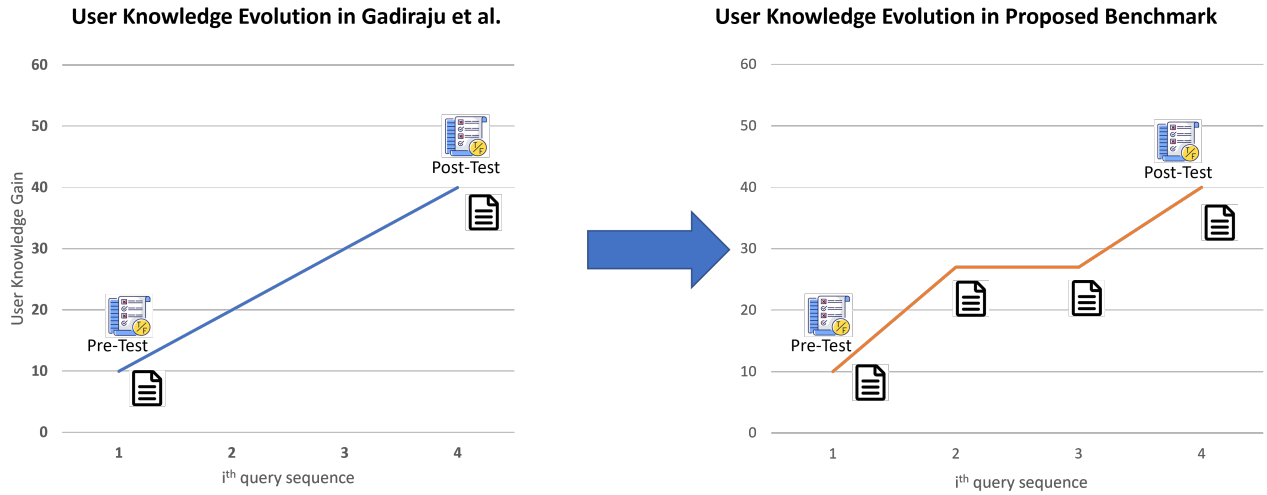

Figure 6.1 – Comparison of user knowledge evolution between the used dataset and the proposed benchmark.

find this dataset a suitable resource for constructing a benchmark to evaluate retrieval algorithms' performance in search as learning contexts.

We filtered the dataset to exclude users who did not interact with the search system, i.e., those who did not visit any page or enter a query. We also excluded users with negative gain, meaning their post-knowledge test score was lower than their pre-knowledge test score. After filtering, the resulting benchmark consisted of 404 users across the 10 topics.

6.3 Benchmark construction

The benchmark we propose tracks the evolution of users' knowledge gained throughout the search session. In this section, we outline the process of constructing this benchmark using the provided dataset. We also introduce a measure g_i that quantifies the contribution of a single document d_i to a user's knowledge gain.

6.3.1 Document identification and text retrieval

We assigned a unique identifier, also referred to as a *urlID*, to each unique visited website or document. The *urlID* was generated in sequential order, starting from *d1* and incrementing by one for each subsequent document.

As discussed in Section 5.4.5.2, we obtained the textual snapshots of the visited documents using the Wayback Machine API. We saved these texts along with their corresponding urlID under the *Corpus* folder in the dataset.

6.3.2 Measuring document knowledge gain

The user's knowledge gain (KG) is the learning outcome resulting from interactions with the search system. By the end of a search session, the user has submitted a set of queries (Q), visited a set of documents (D), and acquired a knowledge gain (KG). The task of this section is to estimate the gain (g_i) brought by a document ($d_i \in D$) visited for a time (t). We refer to g_i as the *document knowledge gain*, which will allow us to estimate the evolution of any user's knowledge throughout any arbitrary session. We define g_i as follows :

Définition 6.3.1 (Document knowledge gain)**.** The document knowledge gain g_i refers to the contribution of a document d_i towards the user's knowledge gain (KG) for each minute spent reading.

The users' answers to the individual test questions were not provided in the original dataset. Therefore, it was not possible to specify the user's knowledge change on every question. Instead, we had correctness test scores reflecting the knowledge states on a specific topic before and after the search session. The knowledge gain KG is the post-session score minus the pre-session score.

Previous work have estimated the user's learning outcomes using linear regression (X. Zhang et al., 2011) and robust regression (Syed & Collins-Thompson, 2018). The intention in the previous work, however, was to study the features of all the visited documents together (ex. total length of queries or the total number of documents explored) rather than one single document. We formulate our estimation task as a prediction problem and use linear regression to solve it. Our adopted dataset provides a set of users (samples), their corresponding KG (dependent variable), and the time t they spent reading a document d (predictive variables). According to research conducted by Wu *et al.*, there is a strong positive correlation between the time spent reading each document and the actual learning outcomes, which justifies the importance of considering reading time as a relevant factor in predicting the knowledge gain (Wu, Kelly, Edwards, & Arguello, 2012). The user's knowledge gain KG is equal to the sum of products of the visited document knowledge gains and their related visiting times. This is summarized by the following equation :

$$KG = g_0 + g_1 t_1 + g_2 t_2 + \ldots + g_i t_i + \ldots. \tag{6.1}$$

We consider that t_i is the total time in minutes, referred to as "active_time" in the original dataset, spent reading the document d_i. Therefore, t_i is the predictive variable of the model, and g_i is the regression coefficient representing the document knowledge gain g_i of d_i. If the user did not visit a document d_i, the related t_i is equal to zero. The g_0 is the model intercept. The document knowledge gain of each document can be found in the file ⟨topic⟩\docgain.csv

It is intuitive to assume that a document cannot "decrease" the knowledge of a user. The worst it could do is not to provide any relevant information. Hence, we constrain the model's coefficients g to be non-negative. A model was generated for each of the 10 topics.

userID	1	2	3	4
39616594	d1	d6	d11	
39033631	d1	d1	d1	
43694802	d1	d6	d8	d11
35912362	d3			

TABLE 6.2 – Sample of the ARWar\user-docbehaviour file showing the ordered list of visited documents for four users.

The number of documents considered by each regression model (the number of parameters g_i) is on average 25.3 ± 8.92, the average R^2 is 0.38 ± 0.11 and the Mean Absolute Error (MEA) is 12.71 ± 3.77. Given the high number of predictive variables in each model and the relatively small population, we can tolerate such error values. In any case, the models were created to estimate the "relevance"—represented as document knowledge gain—of a document, rather than actually predict the user's precise knowledge gain.

We generated a file ⟨*topic*⟩*docgain* for each model and log the *document knowledge gain* g_i of each document. Those values will serve later to trace an estimate of the user knowledge evolution. Table 6.3 shows a sample of the *Sangre \doc-gain* file in the collection.

6.3.3 Logging users' behavior

We define a user's interaction, or behavior, as the set of (query, document) pairs performed during a search session. Each pair is associated with a visit duration logged in minutes. We used this concept of pairs to associate one document with one query, forming a sequence. If a user visits multiple documents for the same query, each document-visit event is considered a new pair. As found by Liu and Belkin, we suppose that the knowledge gained from a document is directly proportional to the time spent reading it (J. Liu & Belkin, 2010). Queries that did not lead to any visited search results are not included in any pairs. On average, users interacted 2.54 ± 1.77 times and viewed 2.31 ± 1.48 distinct documents.

The collection contains two files that record user behavior : the first file ⟨*topic*⟩*user-docbehaviour* logs the set of documents read by every user, chronologically ordered. The second file is ⟨*topic*⟩*user-querybehaviour* that logs the set of queries submitted by every user, chronologically ordered. The queries are lower-cased and spaces at the *start* or *end* are removed. Table 6.2 shows a sample of the mentioned file for the topic *American Revolutionary War*.

6.3.4 Description of the benchmark files

The resulting benchmark is publicly available * and includes the following information for each of the 10 topics :

— **information_need.tsv** : This file contains the information need presented to users before they initiate their search task. Users were asked to satisfy this need by utilizing the search platform.

— **querydoc_index.csv** : This file includes the set of submitted queries along with the related visited documents, as discussed in Section 6.3.6.

∗. https ://github.com/dimaelzein/Benchmark_SAL_Knowledge_Evolution

— **doc-gain.csv** : This file contains the urlID, the associated document knowledge gain (g_i), and the URL link.
— **test-score.csv** : This file includes the userID, the pre- and post-session test scores, as well as the associated knowledge gain (KG). The score range is between 0 and 100.
— **query-click.csv** : This file contains users' behavior, including their submitted query terms and time, and the resulting visited page. It also includes information such as the visited time and duration (in minutes), rank in the SERP, title, and description of the visited pages.
— **user-querybehaviour.csv** : This file includes the userID and the list of submitted queries, sorted chronologically.
— **user-docbehaviour.csv** : This file contains the userID and the ordered list of visited documents, sorted chronologically.
— **docgain.csv** : This file contains the userID, the topic model intercept, and the gain brought by every visited document, in chronological order, as discussed in Section 6.4.2.1.
— **user-cumgain.csv** : This file contains the userID and the cumulative knowledge gain on every sequence, sorted chronologically. This tracks the knowledge evolution and is considered the benchmark, as discussed in Section 6.4.2.1.
— **Corpus** : This folder contains text files containing the scraped text of web pages. All documents were translated to English. The text versions in the original language are also available.

And for all topics :

— **avggain-perseq.csv** : This file contains the average gain brought to users of the same topic on every interaction sequence, for all topics, as discussed in Section 6.4.2.2.
— **test_items.tsv** : This file contains the pre- and post-session test questions with their correct answers.

These files collectively provide a comprehensive and detailed overview of the benchmark dataset, which can be used to evaluate the performance of retrieval algorithms in terms of facilitating user learning in search as learning contexts.

6.3.5 Example : knowledge gains from documents on the *Sangre* Topic

Example 6.3.1 – Table 6.3 displays a sample of the *Sangre\docgain* file in the collection. The information need that was presented to the users before the search sessions for this topic is as follows :

> In this task, you are required to acquire knowledge about 'Sangre de Cristo' mountain range.

By investigating the content of the documents in the table, we notice that $d5$ and $d12$ explicitly discuss the *Sangre de Cristo Mountains* ; they both have a maximum knowledge gain of 100 ; $d6$ offers travel information for *Sangre de Cristo Mountains* ; with details about the town's nature, population, and culture, it reported a high gain of value of 78.8. The document $d22$ discusses the *Rocky Mountains*, along with other general topics like *North American landforms, North America mountain ranges*, but reported a gain of 0, probably because it was written in the Russian language and did not result in a significant knowledge gain. As for the $d21$, it contained mainly pictures of the mountains with very little textual information ; it also reported a gain of 0. Finally, $d13$ contains religious prayers for *Precious Blood* in the Spanish language, which is another topic than the one in the information need ; one possible cause of having this document is that the literal translation of "Sangre de Cristo" is "Blood of Christ". As expected, this document reported a zero gain. We

urlID	URL	doc. gain g_i
d4	https ://en.wikipedia.org/wiki/Sangre_de_Cristo_Range	4.81
d5	https ://en.wikipedia.org/wiki/Rocky_Mountains	100.00
d6	http ://wikitravel.org/en/Sangre_de_Cristo_Mountains	78.82
d12	http ://www.mtns.ru/mountains/subrange/sangre_de_cristo_range	100
d13	https ://www.ewtn.com/spanish/prayers/oraci%C3%B3n_de_la_sangre_de_cristo.htm	0.00
d21	http ://www.mountainphotography.com/gallery/sangres/	100
d20	https ://en.wikipedia.org/wiki/Blanca_Peak	43.31
d22	http ://too.by/webimage/countrys/nalnd.html	0.00

TABLE 6.3 – Sample of the Sangre\doc-gain file, showing the urlID, the URL address and the related document knowledge gain g.

are not justifying the document knowledge gain for any document; the predicted score of some documents is not intuitively justifiable, as they are not credited with any gain despite discussing the Sangre Cristo Mountains.

6.3.6 Corpus and index reconstruction

The dataset included the submitted queries and the links of the visited websites resulting from the query but did not include the content of the pages. Since the content is essential for constructing user knowledge in this thesis, we decided to reconstruct the corpus to include only the visited documents. This ensures the benchmark is reusable by other researchers.

As discussed in Section 5.4.5.2, we reconstructed the text corpus, restored a snapshot of the documents to the date of the study in July 2017, then extracted their textual content (excluded non-content-related sections ie. references, and ads.). A translated version for non-English documents is also made available using the *Google Translate* feature.

To support the future usability of the proposed benchmark, we constructed a query-document index and made it available in the file *querydoc_index* : a lookup file showing all the submitted queries with their related viewed documents. Researchers using the collection must use the corpus of 420 documents - since their knowledge gain is known - for their experiments to compare with our benchmark. The queries were lower-cased and spaces were removed from the beginning and the end of each query. The total number of queries in the index is 188. Table 6.1 shows a sample of the *ARWar\querydoc_index* file.

6.4 Analysis

We provide in this section some statistical analysis about the tracked knowledge per user and per topic.

6.4.1 Document knowledge gain per topic

We conducted an analysis of the average document gain provided by the documents that users visited for each topic.

First, we calculated the average document knowledge gain across all topics, which is 11.97 ± 26.45. Then, we conducted a separate analysis for each topic and present the results in Figure 6.2.

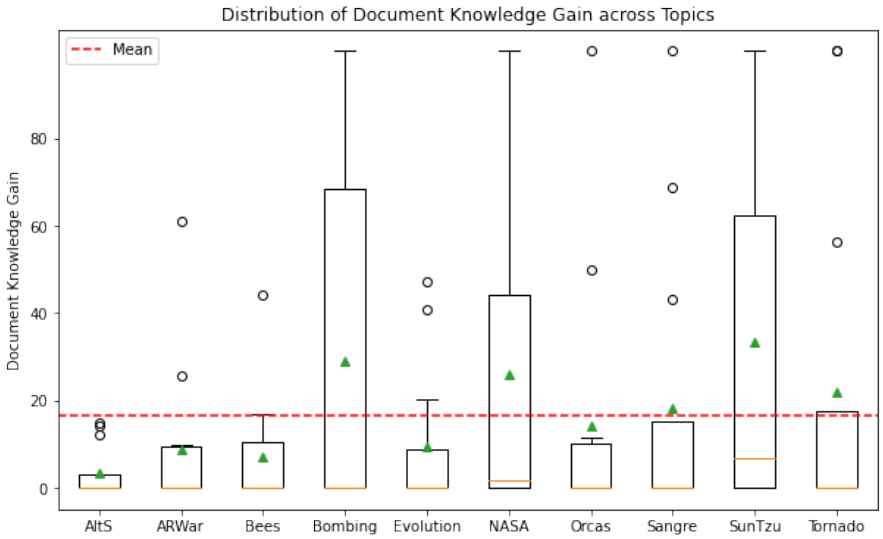

Figure 6.2 – Distribution of document knowledge gain across the topics.

The box plot shows the distribution of document knowledge gain across different topics. The red dashed line represents the overall mean value of the document knowledge gain, which is approximately 18.27. The green dots on each box represent the mean value of the document knowledge gain for each topic. From the plot, it is evident that only two topics, Bombing and NASA, have means higher than the overall mean, indicating that these topics are more informative than the others in terms of document knowledge gain. Additionally, three topics, Orcas, Sangre, and Tornado, have means very close to the overall mean of around 18, suggesting that they are moderately informative. The remaining topics have means below the overall mean, indicating that they are less informative than the others. Therefore, we can conclude that the topics Bombing and NASA are the most informative, while the others are less informative, with some variation in the middle range.

The length of the box in a box plot represents the interquartile range, which is the distance between the 25th and 75th percentiles of the data. Thus, a longer box indicates a larger spread of the data within that class. In this case, three classes - Bombing, NASA, and SunTzu - have longer boxes compared to the other classes, which means that the distribution of document knowledge gain values within these classes is more spread out. This indicates that there is more variability in the knowledge gain across the documents within these classes compared to the other classes.

On the other hand, the other classes have medium to small boxes below 20, which suggests that there is less variability in the knowledge gain across documents within these classes. However, the size of the box alone cannot tell us if the data is skewed or symmetric.

6.4.2 Tracking the knowledge gain evolution

To trace the evolution of the user's knowledge gain KG during their interaction with the system, we consider the set of queries Q they submitted, the set of documents D they visited, the time t they spent reading each document, and the corresponding gain g_i provided by each document.

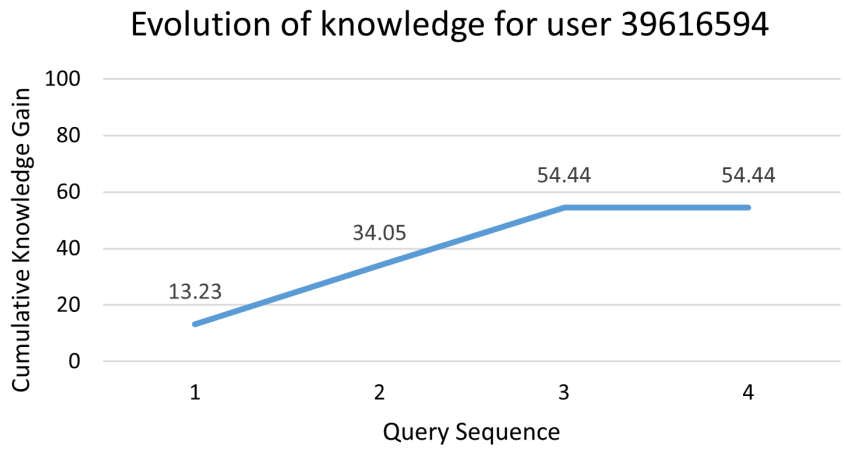

Figure 6.3 – Cumulative knowledge gain evolution for user 39616594 from *American Revolutionary War* topic.

6.4.2.1 Knowledge gain evolution per user

To track how a user's knowledge gain evolved, we rely on their document behavior from the file ⟨*topic*⟩*user-docbehaviour* (containing the set of documents they visited), the time they spend reading each document, and the document knowledge gain g_i in the file ⟨*topic*⟩*docgain*. At the start of the session, we assume the user's knowledge to be the topic's model intercept g_0. Then, for each visited document in the document behavior, we calculate the gain provided by the document by multiplying the knowledge gain g_i with the time spent reading t, and add it to the cumulative gain obtained so far. This enables us to track the user's cumulative knowledge gain and its evolution throughout the session.

The tracked knowledge gain evolution is logged in the ⟨*topic*⟩*user-cumgain* file. The first column 0 shows the topic's model intercept g_0. The subsequent columns show the cumulative gain on every new document read.

For example, Table 6.2 shows that user 39616594, read the list of documents : *{d1,d6,d11}*. The *ARWar\user-querybehaviour* file shows the following set of queries : *{American Revolutionary War, American Revolutionary War, American Revolutionary War}*. The user submitted one query and checked three documents one after the other. The related reading times are 2.20, 0.23, and 0.68 minutes. By multiplying the individual document gain g_i - 9.45, 90.09 and 0.00 respectively for d1, d6, and d11 by the time spent reading in minutes, we obtain the knowledge gain for each sequence. The corresponding values in the *userdocgain* file are $\{20.82, 20.39, 0.0\}$ preceded by the intercept 13.23. The related cumulative gain evolution can be found in the *ARWaw\usercumgain* file; we plot it in Figure 6.3.

6.4.2.2 Average knowledge gain evolution per topic

In this section, we aim to quantify the overall knowledge gained by users on the same topic. To achieve this, we calculate the average document knowledge gain provided to users for every query sequence.

userID	1	2	3	4
25864227	34.91	1		
31511492	2.24	24.39	0	1.15
37953692	11.7	0	5.73	1.95
43691638	16.44	16.44	0	0
...	...	...	...	...
Avg. user KG	**7.488**	**4.219**	**0.751**	**1.478**

TABLE 6.4 – Knowledge gain per sequence and the related average for topic *Tornado*.

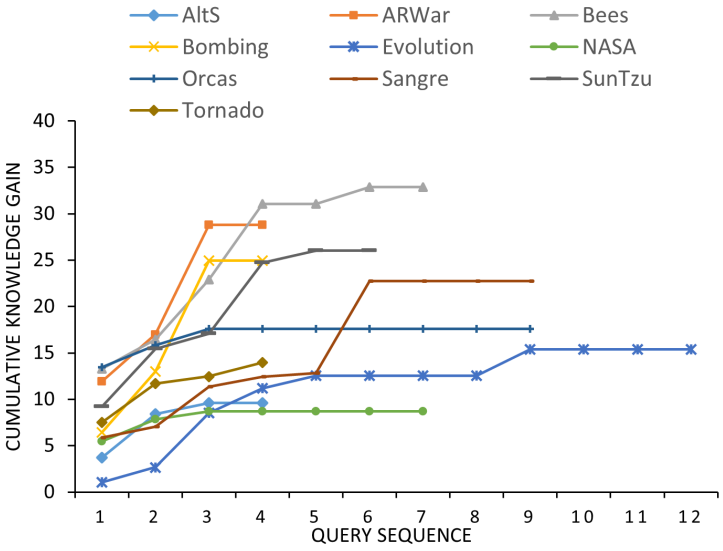

Figure 6.4 – The evolution of the average knowledge per query sequence.

The first few rows of Table 6.4 show a sample of the *Tornado\user-docgain* file, skipping the first column of the file representing the topic model's intercept. The last line shows the average gain for each column. The maximal number of queries submitted (interaction pairs) for this topic was 4. The table shows that users acquired an average gain of 7.488 on their first (query-document) interaction. Users who submitted at least two queries acquired an average gain of 4.219 on their second (query-document) interaction. The calculation of the average knowledge gain is performed for all topics; related results are available under file *Avggain_perseq*. The cumulative average knowledge gain of all the topics is plotted in Figure 6.4.

6.5 Supporting new research directions

The benchmark created for evaluating users' knowledge gain in information-retrieval systems has the potential to facilitate various new research directions in the field. In addition to its primary purpose of evaluating knowledge gain, the benchmark can also be utilized for several other research areas, as outlined below.

6.5.1 Evaluating search systems

The benchmark provides a standardized and well-defined framework for evaluating the performance of information-retrieval algorithms in terms of their ability to facilitate users' knowledge evolution. Researchers can use the benchmark to compare the effectiveness of different retrieval algorithms in improving users' knowledge gain during search sessions. This can help in identifying the strengths and weaknesses of various algorithms and guide the development of more effective retrieval techniques.

Researchers can evaluate a proposed search system using the benchmark by following the steps outlined below :

1. Submit the Set of Queries : Researchers can submit the set of queries used by each user in their search sessions to the system under evaluation. These queries are available in the benchmark dataset files, specifically in the $\langle topic \rangle$*user-querybehaviour* file.
2. Retrieve Query-related Documents : Researchers have two options for retrieving query-related documents. They can either use the list of documents available in the lookup file *querydoc-index*, which is provided in the benchmark dataset, or apply their own search method to retrieve documents from the *Corpus* file, also available in the dataset.
3. Simulate User Selection : Once the documents are retrieved, researchers can simulate the user's selection of a document to read. This can be done by randomly selecting a document from the retrieved list or by using a predefined selection strategy based on the search system being evaluated.
4. Log Selected Document and Knowledge Gain : Researchers can log the selected document and its associated document knowledge gain g_i, which is provided in the $\langle topic \rangle$*docgain* file of the benchmark dataset. This knowledge gain represents the amount of knowledge gained by the user from reading the selected document.
5. Repeat for Query Sequence : The above steps can be repeated for the entire query sequence of the user. Researchers can simulate the user's interaction with the search system by sequentially selecting documents to read based on the retrieved results for each query in the sequence.
6. Trace Cumulative Knowledge Gain and Compare : Throughout the simulated user interaction, researchers can trace the cumulative knowledge gain achieved by the user at each step. This can be done by summing up the document knowledge gains obtained from the selected documents in the query sequence. The cumulative knowledge gain can then be compared to the benchmark to evaluate the effectiveness of the search system in terms of knowledge evolution.

One potential evaluation metric could be the number of sequence queries needed for the user to reach their maximal knowledge gain. The search system under evaluation can be tested to determine if it allows users to achieve their search goals faster than a baseline or other retrieval algorithms. This can provide insights into the efficiency and effectiveness of the search system in facilitating knowledge gain for users, based on the benchmark dataset.

6.5.2 Creating new benchmarks

The methodology used to manipulate the original dataset and create the benchmark has been described in Section 6.3. This methodology can also be applied to other datasets that contain

the following elements : (i) user's knowledge gain, typically assessed through pre- and post-knowledge assessment tests, (ii) search behavior logs, including the sequence of queries and visited pages, and (iii) the searched corpus. While previous research studies have generated datasets that include these elements, their focus has primarily been on the overall knowledge gain after the completion of the search session. By replicating the steps followed in creating this benchmark, researchers can potentially contribute to new research directions.

Furthermore, another possible direction for future research is to use additional datasets to extend this benchmark. This can involve including more users, queries, and documents from diverse domains or contexts. By incorporating data from different sources, researchers can enhance the benchmark's diversity and applicability, allowing for a broader range of evaluation scenarios and enabling the development of more robust and comprehensive information retrieval systems.

6.5.3 Analyzing the factors affecting document knowledge

Given that the dataset now contains the knowledge gain for each document, some analysis can be performed to understand the text features that made the document highly relevant for learning or not. Features such as vocabulary difficulty, text length, number of illustrations, and more can be studied to determine their influence on users' learning outcomes. For example, researchers can explore if documents with simpler vocabulary or shorter text lengths are more effective in helping users learn compared to documents with more complex vocabulary or longer text lengths. This analysis can provide insights into the optimal document characteristics for supporting user learning and can help improve information retrieval systems and user satisfaction.

6.6 Limitations

One of the primary limitations is that unlike other datasets used for evaluations, we do not include a relevance judgment set that considers the relevance of every document with respect to a user's need. While this can be considered a limitation, we argue that this type of information cannot be obtained unless an exhaustive experiment has been conducted to measure the user's knowledge before and after reading every document. The dataset provides an estimation of the gain that a user could potentially acquire from a document, assuming that the document is relevant to their information need.

Another limitation is that the relevance of a document is still considered static for all users, regardless of their background knowledge and cognitive states. In reality, the relevance of a document may vary significantly depending on the user's knowledge and understanding of the topic. For instance, a document that may be useful for a novice learner may not provide any significant gain for an expert learner. Therefore, the proposed dataset should be viewed as a benchmark that represents what an "average" user would see when interacting with a document. This may not accurately reflect the *real* experience of all users, and it is important to consider the limitations of the dataset when interpreting the results of any evaluations. Despite these limitations, the proposed dataset provides a significant advancement in evaluating retrieval algorithms in facilitating user learning and provides a starting point for future developments in this field.

6.7 Conclusion

We have proposed a collection that includes a set of documents and an estimate of the knowledge that users could gain from reading them. To create this benchmark, we utilized a publicly available dataset that tracked the search activity of 500 users across 10 different information needs and multiple topics. We quantified the users' knowledge state before and after their search sessions. We manipulated the data to estimate the knowledge contributed by each document, thereby tracking the evolution of each user's knowledge document by document.

The resulting tracked knowledge evolution serves as a new benchmark that can be used to evaluate the performance of information retrieval systems. Users' knowledge evolution while utilizing these systems can be compared to the benchmark to assess their effectiveness. Additionally, the benchmark can be used to compare the number of interactions or documents required to achieve a search objective. This benchmark will contribute to a better understanding of how users acquire knowledge from individual documents and the determinants "usefulness".

The benchmark includes a set of user behavior data, including questions and clicked results, as well as the evolution of their knowledge throughout the search session. It is publicly available for further research and evaluation purposes, opening new avenues for exploration and advancement in the field.

CHAPTER 7

Extending Jason Programming Language for Knowledge Aware IR

The Belief-Desire-Intention (BDI) architecture, grounded on Michael Bratman's philosophical theory, is a popular approach used for constructing rational agents. Rational agents are those that are endowed with a model of human practical reasoning. In the field of Information Retrieval, a BDI agent can develop its beliefs about the user's knowledge and continuously update this knowledge through decision-making and actions. To achieve this, the user's knowledge is represented as agent beliefs, which are associated with a degree of certainty and revised in case of contradiction or inconsistency. Since agents have some reasoning capabilities, it is possible to use these capabilities to reason and derive new beliefs about the user's agent using knowledge rules. In this regard, we have extended the Jason language, which is a widely-used programming language for constructing intelligent agents with the BDI architecture, to incorporate these functionalities. This chapter details the new features proposed for the extended Jason language and their implementation using IR techniques, along with providing running examples for illustration.

7.1 Introduction and motivation

The field of information retrieval has witnessed significant advancements with the emergence of intelligent agents capable of processing and interpreting large amounts of data. One popular approach in designing such agents is the Belief-Desire-Intention architecture, which models how agents make decisions based on their environment, beliefs, desires, and intentions. In the context of IR, a BDI agent can develop its beliefs about the user's information needs and strive to fulfill those needs through decision-making and actions. Early contributions to agent-based architectures in IR (Guttman & Maes, 1998 ; Bakos, 1997) focused on tracking user activities to personalize web search. Subsequent research incorporated user-related context, such as location and device type, to better understand user behavior during search (Carrillo-Ramos, Gensel, Villanova-Oliver, & Martin, 2005 ; Kurumatani, 2004). The concept of treating users as cognitive agents with their own unique beliefs and knowledge of the world presents a potential solution for customizing search results to each individual's preferences, as suggested by studies such as those conducted by Mora and Rao (da Costa Móra, Lopes, Vicari, & Coelho, 1998 ; Rao & Georgeff, 2001).

We propose a framework that integrates an information retrieval system that is aware of the user's knowledge and needs within a convenient BDI structure. This requires an agent programming language that supports several key functionalities. Firstly, the language must allow for the representation of the user's knowledge as agent beliefs. This means that the agent should be able to store and manipulate beliefs about the user's knowledge state. Secondly, the language should allow for the association of weighted beliefs with a degree of certainty, confidence, or entrenchment. Thirdly, the language must support the revision of beliefs in case of contradiction or inconsistency, especially knowing that the user's knowledge will be frequently updated with new information. the language should enable the updating of the degree of belief if it already exists. As the user's knowledge may evolve over time or new evidence may emerge, the agent should be able to update the degrees of entrenchment associated with existing beliefs to reflect the changing nature of the user's knowledge. Finally, the language must support the derivation of new beliefs through reasoning with knowledge rules. The agent should be able to use its reasoning capabilities to deduce new beliefs from existing knowledge rules, even if the user has not explicitly communicated them. This allows the agent to infer implicit knowledge and provide more personalized and relevant information retrieval results to aid the user in their learning process.

This chapter tackles the following research questions :

RQ7-1 What are the key functionalities required in a programming language to construct an intelligent agent with the Belief-Desire-Intention architecture that can effectively represent the user's knowledge and reason about it for information retrieval ?

RQ7-2 How can we represent a user's knowledge state as agent beliefs in the BDI architecture ?

RQ7-3 How can the changes in the user's knowledge be reflected in the agent belief set ?

RQ7-4 How can we enhance knowledge representation with reasoning capabilities to enable the agent to develop more accurate beliefs about the user ?

We have chosen the Jason language, a widely used programming language for constructing intelligent agents with the BDI architecture, as the framework for implementing our system. The Jason language is renowned for its adaptability and expandability, making it well-suited for incorporating the functionalities needed for our agent. Nevertheless, some of the desired features cannot be implemented using the existing versions of the Jason language. Hence, we intend to extend the Jason language to meet the requirements for the aforementioned functionalities.

This chapter introduces the extension of the Jason language and outlines the new features that are being proposed. It then proceeds to describe the implementation of these features with IR techniques.

7.2 Background

7.2.1 Belief-Desires-Intention agents

The Belief-Desire-Intention architecture is a cognitive architecture used for modeling and simulating intelligent agents that can reason, plan, and act in dynamic and uncertain environments. Three core components are modeled as the agent's mental state : beliefs, desires, and intentions.

— *Beliefs* represent an agent's perception of the environment, including facts, observations, and beliefs about the beliefs of other agents. The set of beliefs is held in a *belief base* that is updated throughout the agent's interaction with the environment.

— *Desires* represent an agent's motivational states. The agent's desires can be transformed into goals that can either be explicitly provided to the agent or learned by the agent from the environment.

— *Intentions* typically represent a set of actions or plans that the agent intends to execute. They are formed through a process of deliberation, where the agent reasons about its beliefs and desires, and selects the most appropriate set of actions to achieve its goals.

The BDI architecture offers a dynamic and adaptable framework for integrating these three components and empowering intelligent agents to reason and generate actions necessary to achieve their objectives. The framework allows agents to modify and update all their components as required in response to changes in the environment.

7.2.2 Rule-based agents

Rule-based agents, also known as rule-based systems, are computer programs that make decisions or take actions based on a set of predefined rules. These rules are typically written in the form of conditional statements, also known as "if-then" statements, and are designed to guide the behavior of the agent in specific situations.

A typical rule-based agent (Jensen & Villadsen, 2015) has a set of ground literals representing facts, and a set of rules in the form of Horn clauses. A literal denoted by α may be optionally preceded by a negation symbol, $\neg$, to indicate that it represents the negation of α, which is denoted as $\neg\alpha$. These literals can originate from various sources such as communication with other agents, observations of the environment, or information obtained from external resources. Each rule in the agent's rule set, has the following form :

$$\alpha_1 \& \alpha_2 \ldots \& \alpha_n \rightarrow \beta \tag{7.1}$$

where the literals $\alpha_1, \alpha_2, \ldots, \alpha_n$ (with $n \geq 1$) are premises of the rule, and β is the *derived* literal. The logical AND operator $\&$ is used to connect the premises in the rule.

When the rules are used in a BDI agent, the literals represent the agent's beliefs stored in the belief base. As the agent performs its reasoning process or adds and removes other facts from the belief base, these literals may undergo changes over time.

7.2.3 Belief revision

7.2.3.1 The AGM belief revision theory

Belief revision is, by definition, the process of modifying the belief base to maintain its consistency whenever new information becomes available. The AGM (Alchourrón, Gardenfors, and Makinson) belief revision theory is a framework that provides a systematic way of updating or revising a set of beliefs or knowledge in the face of new information (Alchourrón, Gärdenfors, & Makinson, 1985). The framework establishes postulates that a rational agent must satisfy when engaging in belief revision. In this theory, a belief base, denoted as K, is assumed to be closed under logical consequence.

Consider a belief base K and a new piece of information α. K is considered inconsistent if both α and $\neg\alpha$ are contained in the logical consequences (Cn) of K, or if $Cn(K)$ is equivalent to $\perp$ (i.e., a contradiction), or if both α and $\neg\alpha$ are logical consequences of K. The framework considers three operators :

— *Revision* $K * \alpha$: adds a belief α to K as long as it does not result in a contradiction within K. If adding α would cause inconsistencies in K, the revision operation starts by making the minimal changes to K necessary to resolve the inconsistency and then adds α.

— *Expansion* $K + \alpha$: adds a new belief α to K without contradicting the existing beliefs.

— *Contraction* $K \div \alpha$: removes a belief α from K along with any other beliefs that logically imply or entail α.

The contraction operation of a formula α from a belief base K aims to produce a belief base that is maximal while not implying α. Let's denote the contraction operator as $\div$. For any given formulas α and ψ, the contraction operator $\div$ should satisfy the following properties :

(K÷1) The result of a contraction, $K \div \alpha$, is a theory (*Closure*)

(K÷2) $K \div \alpha \subseteq K$ (*Inclusion*)

(K÷3) If $\alpha \notin K$ then $K \div \alpha = K$ (*Vacuity*)

(K÷4) If $\nvdash \alpha$ then $\alpha \notin K \div \alpha$ (*Success*)

(K÷5) If $\alpha \in K$ then $K \subseteq (K \div \alpha) + \alpha$ (*Recovery*)

(K÷6) If $\alpha \equiv \psi$ then $K \div \alpha = K \div \psi$ (*Extensionality*)

$K\div1$ ensures that the result of a contraction is a belief base (theory); $K\div2$ ensures that there is no new information added to the belief base after contraction; $K\div3$ ensures that contracting a piece of information that is not believed will not cause changes in the belief base; $K\div4$ ensures the success of contraction which would not work if the piece of information to be contracted was a tautology; $K\div5$ ensures that if α is contracted from K then expanding K with α restores K; $K\div6$ ensures that the result of a contraction is syntax-independent.

The AGM postulates assume that all sentences in a belief set are equally important, which may not always hold true in reality as beliefs can vary in terms of their strength or acceptance level. Williams (M.-A. Williams, 1995b) has proposed a quantitative approach for the AGM framework by introducing finite partial entrenchment rankings to represent the degree of confidence or epistemic entrenchment associated with a particular piece of information. Epistemic entrenchment, as described by Gardenfors (Gärdenfors & Makinson, 1988), captures the notions of significance, firmness, or defeasibility of beliefs, acknowledging that some beliefs may be more entrenched or accepted than others.

Epistemic entrenchment relations create preference orderings of beliefs based on their importance in the context of change. When inconsistency occurs during belief revision, the least significant beliefs (i.e., beliefs with lower entrenchment degrees) are typically relinquished until the consistency is restored. The belief revision operator(s) need to consider the degree of the belief to be added and make a decision on whether to include it or not.

7.2.3.2 Revising and tracking belief by Alechina *et al.*

Preference order The approach introduced by Alechina *et al.* associates a preference order, similar to Williams' approach (M.-A. Williams, 1995a), for each belief and tracks dependencies between them. Beliefs in the proposed approach are associated with preferences, while justifications are associated with qualities. The quality of a justification is represented by non-negative integers ranging from $[0, \ldots, m]$, where m is the maximum size of the working memory. Lower values indicate lower quality, while higher values indicate higher quality.

Définition 7.2.1 (Preference of belief by Alechina et al., 2005)**.** The preference value of a belief α, $p(\alpha)$, is equal to that of its highest quality justification.

$$p(\alpha) = \max\{qual(J_0), \ldots, qual(J_n)\} \tag{7.2}$$

Définition 7.2.2 (Quality of justification by Alechina et al., 2005)**.** The quality of justification J, $qual(J)$, is equal to the preference of the least preferred belief in its support list.

$$qual(J) = \min\{p(\alpha) : \alpha \in \textit{support of } J\} \tag{7.3}$$

Beliefs that are *independent* are typically associated with at least one justification that has an empty support list, which is referred to as a non-inferential justification. These beliefs can originate from the initial belief base or be perceived from the environment. It is assumed that non-inferential justifications are associated with an *a priori* quality.

Belief tracking To track the dependency between the beliefs, the following steps are followed : For every fired rule instance, a Justification J is recorded, which includes a belief α corresponding to the derived belief, and a support list s containing the premises of the rules (contextual beliefs of a plan used to derive α). The dependency information of a belief is represented in the form of two lists : a "dependencies" list that records the justifications of a belief, and a "justifications" list that contains all the Justifications where the belief is a member of support. The approach represents the agent's belief base as a directed graph with two types of nodes : "Beliefs" and "Justifications". A Justification has one outgoing edge to the belief it justifies, and incoming edges from each belief in its support list.

Example 7.2.1 – Figure 7.1 illustrates the belief tracking, considering four beliefs α, β, γ, and μ, and a rule α & $\beta \rightarrow \gamma$. The rule means that if the agent believes in α and β, it believes in γ. For example, Justification J_3 is denoted as (γ,[α, β]) ; γ is the derived belief, and [α, β] is the support list. J_3 is in the *dependencies* list of γ and in the *justifications* list of both α and β. If γ were also derived from μ, i.e. $\mu \rightarrow \gamma$, then its *dependencies* list would also include another justification J_5 denoted as (γ,[μ]). If the belief α was the result of an observation, its dependencies list would include a justification $J_2 = (\alpha, [])$ having an empty support list.

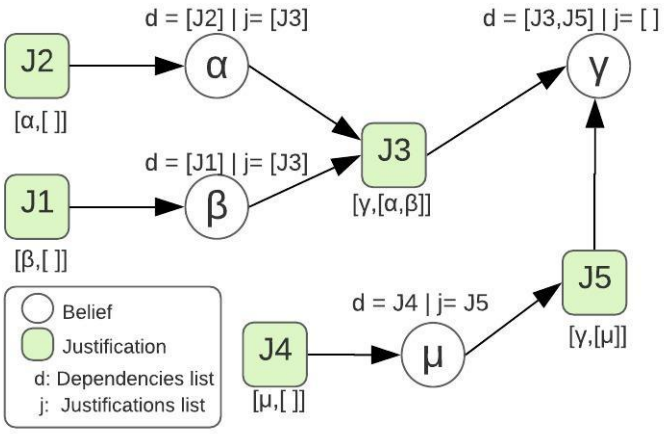

Figure 7.1 – A graph representation of belief dependencies and justifications.

Belief revision and contraction Alechina *et al.* proposed belief revision and contraction operations in their work (Alechina, Jago, & Logan, 2005). These operations are efficient in terms of computation cost, with a time complexity of linear time in the size of the agent's belief base, and they satisfy all the AGM postulates except for (K÷5), the recovery postulate. The authors considered a resource-bounded agent with a finite state and a finite program consisting of a fixed number of rules used to derive new beliefs from the agent's existing beliefs. To manage the complexity, they simplified the language and logic of the agent.

The above equations allow the identification of the weakest member $w(s)$ or "preferred contraction" of a support list s i.e. the member with the smallest preference. The contraction was defined by a literal α as the removal of α and sufficient literals (the least preferred one) so that α is no longer derivable.

Algorithm 7.1 shows how beliefs are contracted in (Alechina et al., 2005), α is the belief to be contracted, β is a derived belief from α, J is the justification, s is the support list and $w(s)$ is the weakest member of the support list. The algorithm 7.2 shows the revision proposed in (Alechina et al., 2005).

Input : Belief Base, α
Output: Belief Base after contraction by α
for *each* $J = (\beta, s)$ *in* α*'s justifications list* **do**
  remove J from β's dependencies list remove J from the justifications list of each literal in s
end
for *each* $J = (\alpha, s)$ *in* α*'s dependencies list* **do**
  if $s == []$ **then**
    remove J
  else
    contract by the literal $w(s)$
  end
end
delete α

Algorithm 7.1 – Belief contraction algorithm by Alechina et al., 2005.

In the example 7.2.1, if we want to contract γ, we have to contract μ (since it is the only member of J_5's support list) and the least preferred member of J_3's support list (either α or β) so that γ is not derivable again.

Input : Belief Base, α
Output: Revised Belief Base
Add α to Belief Base apply all matching plans
while *Belief Base contains a pair* $(\beta, \neg\beta)$ **do**
  contract by the least preferred member of the pair
end

Algorithm 7.2 – Belief revision algorithm by Alechina *et al.*, 2005.

7.3 User knowledge-centric IR agent

We propose in this section a general information retrieval system that extracts the content of documents read by the user to infer their knowledge. The IR system should be equipped with an agent that has beliefs about the user's knowledge, can reason about this knowledge, derive new beliefs, and maintain consistency. The purpose is to use this user-context to personalize future returned documents.

Our proposed framework has the following requirements :

1. **User's knowledge as agent's belief :** The user's knowledge must be represented in a belief base using predicates, which should be expressive enough to convey facts that the user believes is not true. The presence of α in the belief base means that the agent believes that the user knows that α is true.
2. **Negated knowledge is represented :** On the other hand, if the belief base contains $\neg\alpha$, the agent believes that the user knows that α is not true. When neither α nor $\neg\alpha$ is present in the belief base, the agent remains uncertain about the user's knowledge regarding α.
3. **Beliefs are entrenched :** The beliefs should be assigned an entrenchment degree to measure the degree to which the agent believes the user is knowledgeable about a particular

fact. The degree ranges from 0 to 1, where 0 represents the lowest degree, indicating that the agent believes the user has absolutely no knowledge about the belief, and 1 represents the highest degree, indicating that the agent believes the user has maximum knowledge about the belief.

4. **Agent can reason :** The IR agent should possess reasoning power using knowledge rules, which allow the agent to make assumptions about the user's knowledge without it being explicitly stated. New beliefs will be added to the agent represented as derived beliefs.
5. **Beliefs are consistently tracked :** The explicit and derived rules, along with their derivation, should be tracked. The IR agent is modeled as a rule-based entity with knowledge rules that assist in reasoning and deriving new beliefs. During the agent's reasoning cycle, the validity of the rules is checked, and a rule is considered *valid* if all the conditions in the premises are satisfied, the rule is fired, and the belief in the body is added as a new belief. The rules are considered static, and their origin or extraction is not discussed. The agent must have belief revision capabilities.

To maintain the belief base consistency, the entrenchment degree of beliefs must be raised or lowered via a belief revision operation $K * (\alpha, i)$ where α is a new belief and i is its new entrenchment degree. We propose the following to revise belief :

$$K * (\alpha, i) = \begin{cases} \text{If } \alpha \notin K: \ K + (\alpha, i) & \\ \text{If } \alpha \in K: & \\ \quad K + (\alpha, i), & \text{if } i > j \\ \quad Nothing, & \text{if } i < j \\ \text{If } \neg\alpha \in K: & \\ \quad K \div (\neg\alpha, j) \ then \ K + (\alpha, i), & \text{if } i > j \\ \quad Nothing, & \text{if } i < j \end{cases} \tag{7.4}$$

The revision operator checks first if α already exists in the belief base. If it is not in the belief base, it is added with the degree i. If α already exists, their two degrees i and j are compared. When the new degree i is smaller than the existing degree j, the degree of α in the belief base is not changed. When i is higher than the existing degree j, an expansion operation $K + (\alpha, i)$ is initiated and the degree of α is increased from j to i. The revision operator finally checks if $\neg\alpha$ is already in the belief base. If it already exists with a degree j, the preference will be given to the belief with the higher degree. When i is higher than j, α will have the preference to stay, $\neg\alpha$ must be first contracted (or assigned the lowest entrenchment degree equal to zero for example). Then, α is added with degree i. Finally, when i is smaller than j, the addition of α is discarded.

7.4 Jason : properties, limitations & extension

In this section, we present an overview of the Jason programming language architecture (Bordini, Hübner, & Wooldridge, 2007) and its features (version 2.4) (*Jason Agent Programming*, 2021). Additionally, we will address the limitations of the Jason language that prevent us from implementing the desired IR framework outlined in Section 7.3.

7.4.1 Jason : an agent programming language

Jason is an agent programming language designed for developing intelligent agents. Jason is a popular and widely used platform for building multi-agent systems, which are systems composed of multiple interacting autonomous agents that can perceive their environment, reason about it, and take actions to achieve their goals. The *AgentSpeak* language, on which Jason is based, is a variant of the BDI model, which is a well-known and widely used model for agent programming.

The Jason agent language provides a rich set of constructs and features that allow developers to specify the behavior and logic of intelligent agents in a declarative and modular way. It is designed to be human-readable and expressive, making it easy for developers to specify the cognitive processes, decision-making, and interaction patterns of intelligent agents.

7.4.2 Architecture

A Jason agent, similarly to other agents modeled in BDI, is defined by sets of *beliefs*, *plans*, and *goals or intentions*. In this section, our primary focus will be on presenting the syntax of Jason language for expressing beliefs and plans, as these elements will be central to our discussions in subsequent sections.

7.4.2.1 Beliefs

Beliefs in Jason are represented by predicates, which are used to express the agent's beliefs about the world. The presence of a predicate in the agent's belief base means that the agent currently believes it to be true. For example, a belief α represents that the agent believes the predicate α to be true. The $\sim$ operator in Jason is used to represent negation, explicitly indicating that the agent believes a literal to be false. For example, $\sim \alpha$ represents that the agent believes the predicate α to be false.

Annotations are used in Jason to list some additional details and metadata about beliefs such as their source, reliability, or temporal information. This list of information is placed after a belief, enclosed in square brackets. For example, a belief with an annotation could be written as "[source(a), time(t), p]" where "source(a)" indicates the belief's source is "a" and "time(t)" indicates the belief was formed at the time "t". The "source" annotation is a special standard annotation that automatically records the name of the source from which the information in a belief was obtained. This annotation has a predefined meaning and is interpreted by the Jason system, allowing the agent to keep track of the sources of its beliefs in a transparent and understandable manner.

7.4.2.2 Plans

Plans in Jason are used to represent the agent's intended course of action to achieve its goals. Plans are written in a declarative manner of a set of rules specifying the sequence of actions or events that the agent should take in order to achieve its objectives.

A Jason plan is a rule composed of three parts : the *triggering event*, the *context*, and the *body*. It is expressed as follows :

$$+triggering\ event : context \leftarrow body. \tag{7.5}$$

```
/* Beliefs */
salesUp(company1).
trust(marketstocksite).

/* Plans */
@p1 +salesUp(X)[source(S)] : wellManaged(X)
                & trust(S) <- +goodToBuy(X).
```

Figure 7.2 – Snapshot of Trader Jason agent's initial state : situation A

triggering event represents one condition that might activate the plan ; it can be the addition (+) or deletion (-) of a belief or a goal. *context* is a conjunction of literals that need to be satisfied to make the plan applicable – and possibly executed. A plan is applicable if : (i) first, its triggering event occurred, and (ii) its context (one or several conditions) is a logical consequence of the agent's current beliefs. Together the triggering event and the context constitute the plan's *head*, they define the conditions under which the plan should be executed. The *body* is a sequence of actions or *goals* to be performed when those conditions are satisfied.

7.4.3 Constraints

We will explain in this section the constraints in the publicly available version of Jason that limit us from implementing our desired framework. We will illustrate this through a running example.

7.4.3.1 Example : Stock Trader agent

We propose a modified version of the *Stock Trader* agent example presented in (Alechina, Bordini, Hübner, Jago, & Logan, 2006), as a running example in the rest of the chapter. We will use this example to illustrate the Jason syntax, its properties, and limitations, as well as our new proposed features later.

Example 7.4.1 – We consider a *Stock Trader* agent that communicates with other agents to receive financial information and has access to Web Services that provide news about the stock market. The agent is trying to decide which stocks to buy or sell, based on its existing beliefs and the information received or perceived. We present two initial situations for the *Stock Trader* agent, along with the related operations to be executed. The operations for each situation are executed sequentially. In this section, we will perform the *operations* on the given *situations* to demonstrate the limitations of the original Jason version in the IR framework we aim to improve. Later in Section 7.5, we will repeat the same operations to show how our proposed extension overcomes these limitations

Situation A The *Stock Trader* agent is represented by its initial state in Figure 7.2. It has two initial beliefs and one plan. When the plan *p1* is executed, it will result in the addition of the belief goodToBuy. For this situation we only have one operation :

— **Operation A.1** : Add the belief *wellManaged(company1)* after Agent *Ag2* informs *Trader* in *Situation A* that the *company1* is wellManaged.

```
/* Beliefs */
wellManaged(company1).
trust(marketstocksite).
limits(company2,30).
price(company2,50).

/* Plans */
@p1 +salesUp(X)[source(S)] : wellManaged(X)
                & trust(S) <- +goodToBuy(X).
@p2 +salesUp(X) : ~wellManaged(X)
                        <- +sellStocks(X).
@p3 +salesUp(X) : True <- +watchlist(X).
@p4 +price(X,Z) : limits(X,Y)
                    & Z>Y <- +sellStocks(X).
```

Figure 7.3 – Snapshot of Trader Jason agent's initial state : situation B

Situation B The *Stock Trader* agent is represented by its initial state in Figure 7.3. The agent believes that *company1* is well-managed and trusts the source *marketstocksite*. It has set a limit order to sell stocks of *company2* for 30 euros, while the current market price for the same stock is 50 euros. The agent's plan library contains four plans, the first three of which may be executed upon being informed of increased sales of a company, while the fourth plan will be executed if the current price of a company's stock has reached or surpassed the limit order. In *Situation B*, the interpreter will execute plan *p4* since it is the only plan that satisfies its conditions, so far. As a result, the belief that the agent should sell stocks of *company2* (*sellStocks(company2)*) will be added to its belief base.

We now describe the operations to execute for Situation B :

— **Operation B.1 :** Add salesUp(company1). The agent *marketstocksite* informs *Trader* that *salesUp(company1)*.

— **Operation B.2 :** Add ∼wellManaged(company1). An agent *Trader* that a crooked CEO has been fired from company1.

— **Operation B.3 :** Add ∼trust(marketstocksite). A trusted Web service broadcasts marketstocksite is not trustworthy anymore.

— **Operation B.4 :** Add ∼sellStocks(company2). A trusted Web service informs holding stock sell for company2.

We note that both *Situation A* and *Situation B* share the same plan *p1*. The distinction lies in the agent's initial belief : in *Situation A*, the agent initially believes that the sales of *company1* are increasing and later receives information about the company's good management in Operation A.1, whereas in *Situation 1*, the agent initially believes that *company1* is well managed and later acquires information about the sales increase in Operation B.1.

7.4.3.2 Trigger event and plans' execution

The structure of Jason's plans in Equation. 7.5 is reliant on the triggering event's occurrence. A plan is executed only if the conditions in its context are satisfied before its triggering event occurs. The order in which the triggering event and the context conditions are specified matters for plan execution. For example, the plan *p1* will not be triggered in **Operation A.1**; but triggered and executed in **Operation B.1** resulting in the addition of *goodToBuy(company1)[source(self)]* *.

∗. Source annotation will be omitted for the rest of the Chapter.

Knowing that our aim is to use Jason's plans to represent knowledge-rules; we try to represent *salesUp(X)[source(S)]* & *wellManaged(X)* & *trust(S)* $\rightarrow$ *goodToBuy(X)* : the plan *p1* should be replaced by three plans with three triggering events corresponding to the three literals (*salesUp*, *wellManaged*, *trust*).

In order to represent a knowledge-rule, it is necessary to have a set of plans that is equivalent to the number of literals present in the context, plus one additional plan for the trigger condition.

7.4.3.3 Option selection

In Jason, it is possible for multiple plans to have the same triggering condition but different contexts. These plans are categorized as *relevant plans* when the triggering event occurs. Jason then examines the context of each *Relevant plan* and stores those that satisfy their context in a list called *applicable plans* or *Options*. The *selectOption* function is used to choose one plan for execution, with the default behavior being to select the first option based on the order in which plans were written in the agent code. However, this limitation of Jason's plan syntax restricts its ability to represent knowledge-rules. In situations where two or more rules have the same trigger and all conditions of the rules are satisfied, only one rule will be fired, which can result in the loss of important information.

In the Trader example, we would expect both *p1* and *p3* to be executed in case of a price decrease and sales increase. However, with the current version of Jason, it is not possible to represent knowledge-rules in a way that both plans can be executed. This can be a significant drawback in situations where multiple rules need to be triggered based on the same condition, especially in domains where the correct interpretation of knowledge-rules is needed.

7.4.3.4 Belief base consistency and preferences on beliefs

The creators of the Jason language were aware of the potential need for belief revision and have provided the infrastructure with the *brf* function for this purpose. However, it should be noted that the default implementation of this function does not perform any belief revision operation unless overridden by the programmer/user with a specific implementation.

The default *brf* function in Jason, which takes parameters *Literal belieftoadd* and *Literal belieftodelete*, simply updates the belief base with the literals to be added or deleted, without verifying the consistency of the belief base. This means that contradictory beliefs may be accepted in the belief base without any preference or resolution. The customization of the *brf* function to include belief revision was left to be done by the programmers.

Returning to our example, the beliefs resulting from **Operations B.2, B.3, and B.4** will be added to the belief base, regardless of whether or not they create inconsistencies within the base.

7.4.3.5 Belief certainty

In the Jason programming language, the concept of belief is treated in a Boolean manner, meaning that an agent either believes something to be true, false or is completely unaware of it. However, in order to allow for a more nuanced representation of an agent's belief, Bordini and colleagues introduced the concept of a belief's "Degree of Certainty" in their 2007 paper (Bordini et al., 2007). They proposed an agent named *Maria* who believes another agent named *Bob* is colorblind with a certainty of 0.7, this belief can be expressed as $colorblind(bob)[source(self), degOfCert(0.7)]$. However, it should be noted that the "degOfCert" annotation does not have a predefined meaning

Feature	Original	Extended
Beliefs		
Dependencies	Not tracked	Tracked
Inconsistency	Accepted	Not accepted
Graded (degOfCert)	No	Yes
Preference New $\succ$ Old †	No preference	High $\succ$ Low
Plans		
Knowledge-rule with n conditions	n plans	1 plan
Order of conditions	Dependent	Independent (with +tei)
Execution of applicable plans with same triggering event	One plan only	All plans

TABLE 7.1 – Comparison between Jason and its extension's features.

for the Jason interpreter. Instead, it is up to the programmer to define and interpret the concept of certainty in a way that is appropriate for their specific application.

7.5 Extending Jason with graded belief revision capabilities

We propose an extension to Jason that aims to efficiently revise graded beliefs. Our objective is to model rule-based agents using Jason, considering beliefs as facts and plans as rules. To achieve this, we first proposed the concept of *trigger-independent plans.* As we highlighted in Section 7.4.3, the existing approach only triggers a plan in the presence of a triggering event. If the conditions for executing a plan were satisfied with a non-triggered literal, the plan would not execute. This approach is not suitable for dealing with belief change.

One of our objectives is to increase the flexibility of the triggering process to enable the representation of knowledge-rules. Additionally, we have implemented the dependency approach proposed by Alechina *et al.* in (Alechina et al., 2005) discussed in Section 7.2.3.2, which was also used in (Alechina et al., 2006) to track the dependencies between beliefs by associating dependency and justification lists with each belief. We have adapted their algorithm to calculate the qualities of justifications and the preferences of beliefs. As a result, we have introduced a graded notion of beliefs, which is represented by the variable *degreeOfCert*, and we have proposed a new algorithm for handling inconsistencies that may arise when a new graded belief is received.

In this section, we introduce our extension to certain Jason features to better suit our knowledge-aware information retrieval goals. We also provide a comparison between the original features and our proposed extensions in Table 7.1.

7.5.1 Representation and execution of knowledge rules

To represent a rule having n conditions in the form of $\alpha_1 \& \alpha_2 \ldots \& \alpha_n \rightarrow \beta$ using Jason plans, the premises of rules are supposed to be in the plan's head. That means that one of the conditions of the premises must be the triggering event and the others in the context (for example, $+\alpha_1 : \alpha_2 \,\&\, \ldots \alpha_n \rightarrow \beta$). However, the plan execution in the original Jason version is reliant

on the occurrence of the triggering event : a plan is executed only if the context conditions were satisfied before the triggering event takes place. The order of the triggering event and the context conditions matter for the execution of a plan. If the condition α_1 was satisfied before the others, the plan will not be executed. One alternative could be to write n plans.

The extended version of Jason allows the expression of knowledge-rules by the so-called *Trigger-Independent plans*. Those plans will be executed whenever the combination of several conditions is satisfied, no matter which condition was satisfied first. In other terms, they do not wait for one specific trigger condition to occur to execute the plan. The syntax of *Trigger-Independent plans* should have the reserved word "+*tei*" that stands for *trigger event independent* in the trigger part and all the other conditions in the context. The plan's new syntax to represent knowledge rules is proposed :

$$+tei : context' \leftarrow body. \tag{7.6}$$

$context'$ has all the premises α_1 & α_2& . . . &α_n and the $body$ contains β the *derived belief*.

Using the original Jason in the case where two or more plans had the same trigger and all had a satisfying context field, would return only one plan for execution. The returned plan would be by default the first plan according to the order in which plans were written in the code. Contrarily, using the extension in the same case would return/execute all the plans having satisfying conditions.

To implement this change we have modified the *addbel* and *brf* functions in the Jason code. We modified the *brf* function to include the "belief" *tei* every time a belief (whether initial, communicated, perceived, or derived) is added using the *addbel* function. This means that all plans will have *+tei* as a trigger event, making them relevant plans whenever any belief is added. Consequently, the interpreter will evaluate the conditions in the contexts and select the plans that have satisfying contexts as either *applicable plans* or *options*

In the example we provided, the application of **Operation A.1** results in the addition of *tei* along with the belief *wellManaged(company2)*, which in turn triggers plan p1. The same would happen if any of the other context conditions (e.g., *trust*, *wellManaged*, or *salesUp*) were added, as all three conditions are present.

We would like to note that users will still have the option to use the original plan syntax and switch between the two syntaxes as needed. This means that the proposed extension is backward compatible.

7.5.2 Degree of certainty

The notion of "believing" in Jason is Boolean : An agent either believes something is true or false or is ignorant about it. We allow the representation of gradual beliefs by expressing "degOfCert(X)" in the annotation part of a belief. Here, X represents the belief certainty defined as follows :

Définition 7.5.1 (Agent belief's degree of certainty)**.** The degree of certainty associated with a belief α indicates the extent to which the agent believes α to be true, and it is expressed as a value between 0 and 1.

The degree of certainty associated with *initial beliefs*, *beliefs communicated* by other agents, and *beliefs perceived* by the agent must be explicitly specified by the source of the information being sent. For instance, an initial belief such as *wellManaged(company1)[degOfCert(0.5)]* can

be defined with a degree of certainty of 0.5. The degree of certainty for *derived beliefs* will be discussed in Section 7.5.3.

7.5.3 Deriving and tracking beliefs

Derived beliefs are dependent on the premises that derived them; therefore to calculate their related degOfCert, the dependency between beliefs must be tracked. We track in this extension the beliefs following the approach discussed in Section 7.2.3.2 : a justification is represented by a derived belief, a support list, and a quality; a belief is represented by a dependencies list, a justifications list and a degree of certainty. Whenever a knowledge-rule, represented by a trigger-independent plan, is fired and a new belief is added, a justification node is created. This node links the rule's premises with the derived belief. The degree of a derived belief is automatically calculated by the interpreter using Equation 7.2. When any of the beliefs are contracted, the related justifications are removed as well. Justifications with empty support lists are created upon the addition of initial, communicated, and perceived beliefs.

Unlike Alechina et al. (Alechina et al., 2005), we do not assign any *a priori* or predefined qualities to justify independent beliefs, as their degrees are explicitly stated. It is worth noting that our approach differs from Alechina's various proposals, which assume that only tautologies can have the highest degree of certainty, 1. Following Dubois and Prade (Dubois, Lang, & Prade, 1991), we assume that other formulas are allowed to have the highest certainty. However, if a new belief arrives with the highest degree of certainty, we have chosen the option of keeping the new belief and discarding the old ones.

Operation B.1 As an example, when *salesUp* is added with a degree of certainty 1, and p1 is executed, *goodToBuy(company1)* will be added. The justification for *goodToBuy* will have a quality equal to 0.5 (equal to the least preferred member in its support list = min(0.5,0.8,1)). Therefore, the certainty of *goodToBuy* will be equal to 0.5 since it has only one justification.

7.5.4 Belief revision

Contradictory beliefs were accepted in Jason's belief base and no belief revision was performed; no preference on beliefs either. The agent could believe in α and its opposite $\sim \alpha$ at the same time. The extended version integrated the notion of belief's certainty into the belief revision decisions and did not allow belief inconsistency. In case of contradiction, the preference is given to the belief with the higher degree : the belief with the smaller certainty degree in the inconsistency pair will be contracted/discarded, and the other belief will be added/kept. In the case of equal certainties, the new belief is given the preference.

A contraction algorithm was proposed : A belief α is not contracted unless a more preferred belief $\sim \alpha$ was added. When contracting a belief α, there is no need to contract beliefs that derived it : when the rule deriving α will attempt to add it again, the addition will be discarded because it will be faced by $\sim \alpha$ which is more preferred. In other terms, the belief in question is contracted with its related justifications without contracting neither the rule's premises nor the rule itself. Beliefs with no justifications will also be contracted.

Our contribution here is two-fold : we have modified the *brf* function to check if the addition of a new belief will cause inconsistency in the belief base, and, we integrate the notion of belief's certainty into the belief revision decisions. The belief with the smaller certainty degree in the inconsistency pair will be contracted/discarded, and the other belief will be added/kept. In the case

of equal certainties, the new belief will have the preference to be preserved. Our implementation gives the developer the flexibility to switch the preference between new and old information.

We propose the following contraction algorithm :

Input : Belief Base, α
Output: Revised Belief Base
foreach $J = (\beta, s)$ *in* α*'s justifications list* **do**
  remove J from β's dependencies list remove J from the justifications list of each literal in s remove J from graph
end
foreach $J = (\alpha, s)$ *in* α*'s dependency's list* **do**
  remove J from the justifications list of each literal in s remove J from graph
end
delete α Remove all beliefs with an empty dependencies list.

Algorithm 7.3 – Proposed algorithm for belief contraction.

Our model does not contract a belief α unless a more preferred contradictory belief $\sim \alpha$ is added. When contracting a belief α, we don't see a need to contract beliefs that derived α : when the rule deriving α will tempt to add it again, the addition will be discarded by the *brf* function because it was faced by $\sim \alpha$ that is more preferred. In other terms, we contract the belief in question and the related justifications without contracting neither the premises of the rules nor the rule itself.

Remark 1. *In comparison with Algorithm 7.1, we observe that our algorithm does not perform the recursive removal of justifications in the second "for loop". Therefore, the complexity of our algorithm cannot be greater than that of Algorithm 7.1.*

Operation B.2 - Add *∼wellManaged(company1)[degOfCert(0.9)]*. When this operation is received, it will be treated by the *brf* function, which will detect inconsistency with the existing belief
wellManaged(company1)[degOfCert(0.5)], and will compare the certainties of the pair. As priority is given to the belief with the higher degree, the *brf* function removes *wellManaged(company1)* first, then adds *∼wellManaged(company1)*. Knowing that wellManaged(company1) was a member of the sole justification of *goodToBuy(company1)* ; the contraction of *wellManaged(company1)* will result in the contraction of *goodToBuy(company1)* as well. Finally, p2 is executed as its conditions are satisfied ; *sellStocks(company1)[degOfCert(0.9)]* is added.
Operation B.3 : Add *∼trust(marketstocksite)[degOfCert(0.7)]*. Similarly to Operation B.2, when inconsistency occurs, the agent prefers to keep the belief with the highest degree. In this case, the addition of *∼trust(marketstocksite)* will be discarded.
Operation B.4 : Add *∼sellStocks(company2)[degOfCert(0.4)]*. *sellStocks* has a certainty of 0.2 in the belief base. When *∼sellStocks(company2)* is added, it would be more preferred than *sellStocks(company2)*. The agent would then contract *sellStocks(company2)* and add *∼sellStocks(company2)*. On the next reasoning cycle, the plan p4 is made applicable and will attempt to add *sellStocks* with a degree of 0.2 again. However, the addition will be discarded because the agent believes *∼sellStocks* with a degree of 0.4 higher than 0.2.

7.6 Discussion and limitations

Our main goal in this chapter was to enhance the capabilities of Jason agents to track and revise the user's knowledge while maintaining a consistent representation. By extending Jason, we were able to adapt the agent to fulfill the requirements presented in Section 7.3. To establish a connection between the user in IR concepts and those in Jason, we provide the following mappings :

Agent's belief about the user's previous knowledge : The initial beliefs of the Jason agent can represent the agent's assumption about the user's previous knowledge. These initial beliefs are the agent's representation of what it believes the user knows or does not know. While this information may not necessarily be true, it can be refined using knowledge rules and belief revision techniques.

Agent's belief about knowledge extracted from documents : The extracted beliefs represent the information that the user absorbed from reading a document during a search session. They are directly extracted from the text and added as a belief of the agent about the user's knowledge, similar to how information is communicated by another agent. To represent a belief α, it must be extracted from some text and could be a keyword, a concept, or an instance of a Uniform Resource Identifier from an ontology for example. These beliefs are keywords that are extracted from the content of the documents, and can be considered as perceived beliefs.

Agent's belief about the user's derived knowledge. Derived beliefs are the beliefs that are inferred by the agent using its knowledge rules, based on the existing beliefs about the user. These beliefs are not directly stated or observed but are derived or inferred by the agent based on its reasoning process. The agent can use its existing knowledge about the user to derive new beliefs and make informed decisions based on the available information.

Beliefs have certainties, knowledge has entrenchment. The proposed extension allows for the representation of the degree of certainty of the agent's belief. This information will be represented by *degOfCert* and logged in the belief annotation of Jason. The degree can represent the strength of the user's knowledge of some concepts. For initial beliefs, the degree of certainty is predefined. For perceived beliefs, their degree is associated with the strength of the extracted information from the text, usually returned by the keyword/concept extraction methods. Finally, for derived beliefs, it will be calculated using Equation 7.2. It will represent the agent's estimation of the user's knowledge regarding a concept.

Negated beliefs and knowledge are represented The original Jason language has a belief syntax that allows for representing negated beliefs using the $\sim$ operator. In a user IR context, this operator represents the agent's belief that the user's knowledge of the information α is not true. While this feature already exists in Jason, our extension made it inconsistent to have both α and its opposite $\sim \alpha$ coexist because they are contradictory.

We would like to mention that extracting negated predicates from the text can be a challenging task. While there are NLP tools available that can perform this task, defining rules for negations can be complex. Alternatively, a machine learning approach, such as training a classifier to predict whether a given sentence contains negation or not, can be used. However, this can add complexity to the system's architecture.

Reasoning with knowledge rules We wanted to adapt Jason to have the capability of representing knowledge rules in the form of $\alpha_1 \& \alpha_2 \rightarrow \beta$. We took advantage of the structure of the Jason plans, but we found that it was not suitable since the plans were triggered by specific limited events. Therefore, we proposed a new variation of Jason plans that had a new syntax to represent knowledge rules, as follows : $+tei : \alpha_1 \ \& \ \alpha_2 \leftarrow \beta$. This allowed us to represent the knowledge rules in a more flexible manner that suited our needs. Although the origin of the rules was not our primary focus, we mention that they can be automatically extracted using available tools and corpora like the information flow method (Song & Bruza, 2003), and then fed to the agent. However, similar to beliefs, some rules may need to be revised, a topic needing further investigation.

Beliefs are tracked We utilized the tracking algorithm introduced by Alechina *et al.* (Alechina et al., 2005) to monitor the beliefs, rules, and their origins, while also taking into account the degree of entrenchment of a belief.

Beliefs are revised Jason, like other agent programming languages, has the capability to perform reasoning tasks, allowing an agent to use available information to draw logical conclusions or make informed decisions. Our aim was to leverage this capability to enable the IR agent to reason about the consistency of its belief base using predefined belief revision algorithms. We modified the Jason functions to include these algorithms, presented in Equation 7.3, which enable the agent to handle new information in three ways : if it already exists in the belief base, if it is entirely new, or if it contradicts existing beliefs. This approach enables us to maintain consistency in the belief base.

Finally, the current framework only considers the user's knowledge and tracking without accounting for their information needs. The BDI structure and the Jason language have the potential to represent the user's needs using their intention and goals structure. However, there is still room for exploration on how to integrate these features into the system effectively. This can include developing mechanisms to represent user intentions and goals, and incorporating them into the belief and desire components of the BDI architecture.

7.7 Conclusion

In this chapter, we introduced the integration of the Belief-Desire-Intention architecture into information retrieval. Our proposed framework consists of an agent following the BDI architecture, which possesses beliefs about the user's knowledge and has reasoning capabilities to infer new beliefs through predefined rules.

The choice of an agent-based structure is based in its ability to represent each user with an agent that manages their knowledge and interacts with another agent : the search engine. The user's agent is responsible for making decision about which documents to present to the user, based on their relevance. Those decisions must serve the agent's objective to help the user learn by comparing the content of the documents to the user's knowledge.

We identified Jason as a suitable agent programming language for this context, thanks to its flexibility and Java-based development, which facilitates customization. We proposed an extension of the Jason language for BDI agents, highlighting its features for representing the user's knowledge, tracking the user's knowledge, and maintaining consistency through belief revision algorithms. Key limitations of the proposed extension were also discussed, such as the difficulty

in extracting negated predicates and the challenges in handling complex and uncertain environments.

Future research directions could focus on the integration of BDI and IR frameworks to effectively represent the user's information needs using intention and goals structures. Additionally, addressing the limitations of the proposed extension of Jason language through the use of machine learning approaches or other rule-based techniques, presents a promising direction for further exploration and development.

CHAPTER 8

Conclusion and Future Work

8.1 Conclusion and discussion

presents the core of our work, where we explore different representation methods for the user's knowledge. We have introduced a comprehensive framework that can extract the information the user acquires from text documents during their session and update their user profile accordingly. We have also de mFuture work could investigate incorporating named entities in a knowledge graph where the relationships between the entities may have significance, and test the accuracy of such structure to capture the user's knowledge state and gain.

In this thesis, we have contributed to the field of search as learning by focusing on the understudied aspect of knowledge and information need representation. By exploring this area, our goal was to explore how these search systems can track their users' progression, and knowledge growth, towards their learning objectives. Chapter 4 presents the core of our work, where we explore different representation methods for the user's knowledge and information needs. We have introduced a comprehensive framework that can extract the information the user acquires from text documents during their session and update their user profile accordingly. First, we have also demonstrated the effectiveness of simple methods, such as tracking the occurrences of the words the user is exposed to when reading, in estimating the user's learning gain in relation to a specific topic objective. We have compared this approach to more complex mehtods, where the documents and the user's knowledge was represented using language models like BERT, or using identified named entities. The results showed that using named entities alone did not result in acceptable performance in estimating the user's knowledge gain. When combined with other representations, they were able to enhance the framework's performance.

Our contributions in evaluating algorithms that aid users in learning can be divided into two parts. The first is an evaluation measure that assesses the relevance of a document based on the user's evolving knowledge during the search session, with particular emphasis on the occurrence of topic-related words. The proposed measure showed its effectiveness, and importantly, our experiments demonstrated that accuracy improves when the user's prior knowledge of the topic is taken into account. This underscores the importance of personalized approaches in search systems, highlighting a shift towards more user-centric models in IR. Our experiments demonstrated that both keyword-based profiles and language model-based profiles exhibit comparable results and maintain a reliable performance even as the search session grows longer and the representation of the user's knowledge becomes more complex. The second contribution in evaluation is a resource dataset that estimates the relevance of a set of documents in terms of their contribution to a user's knowledge gain. The proposed resource provides a tracking mechanism for the evolution

of a user's knowledge gain at a document-by-document level. This dataset is a significant step towards creating more dynamic and responsive IR systems, enabling more precise and user-specific evaluations. This dataset can serve as a benchmark for future researchers who want to compare the performance of their algorithms against a baseline.

Finally, our last contribution was to propose an extension of the Jason agent programming language to give it the capabilities of representing the user's knowledge as agent beliefs and also maintaining the consistency of the represented knowledge. This new version of the language will allow future IR frameworks to include a BDI component capable of modeling the mental processes of a user.

8.2 Limitations and future work

In this section, we examine some limitations of our work and explore potential approaches for addressing these challenges in future research.

8.2.1 Previous knowledge and cold start

One common challenge in the field of personalization and recommendation is the cold start problem, which refers to the difficulty of adapting the search system for new users who do not have any previous interaction history with the system. In our presented work, this difficulty could be twofold : (1) obtaining the user's previous knowledge and (2) representing it in a coherent way with the granular representations used in the frameworks we propose.

The RULK framework presented in Chapter 4 leveraged different representations of the user's knowledge and predicted the learning gain at any time of the session. The framework was designed to account for the user's previous knowledge but did not have a method to extract or predict it upon the first interaction with the system. Furthermore, the experiments and dataset employed in this research were explicitly tailored for individuals with limited prior familiarity with the search topic, and the framework exhibited encouraging outcomes within this scenario. However, it's important to recognize that it's still uncertain how the framework would behave in situations where the user possesses prior knowledge that the framework couldn't account for.

The evaluation measure proposed in Section 5 also accounted for the user's previous knowledge. Experimental findings showed acceptable results when the user's previous knowledge was considered null, and better results when it was taken into consideration. The dataset used did not contain detailed user knowledge prior to the session but was instead represented as a score, which served as an estimate of the user's knowledge. Our experimental results demonstrated satisfactory outcomes in capturing the user's knowledge gain after a session, even when the given previous knowledge was represented as a score.

In future work, it would be valuable to use datasets where users have existing knowledge on certain topics that the framework is not initially aware of. To address the cold start in this case, one can use existing methods from the literature that utilize classification and prediction techniques to obtain knowledge scores or familiarity levels (J. Liu et al., 2016b; R. Yu, Gadiraju, Holtz, et al., 2018b), at the early stages of the session.

8.2.2 Factors influencing learning and information absorption

In our work, we assume that when a user visits a page, they comprehensively absorb all the information it contains. Our proposed research did not differentiate between users based on their information absorption abilities, learning styles, or document preferences. Nonetheless, we acknowledge that both internal factors related to the user and external factors linked to the documents significantly influence the user's learning experience.

We have categorized user-related factors into two distinct groups : cognitive factors and behavioral factors when considering the elements that impact users' learning experiences. The cognitive factors can include the user's motivation, attention span, working memory capacity (Baddeley, 2007), curiosity, reading comprehension (Coiro, 2011), interests in the topic they are learning about (J. Liu & Jung, 2021), and cognitive load. In the longer term, it is essential to consider the aspect of forgetting as well. Over time, users may forget some of the information they've acquired, with varying rates of forgetting. Integrating the concept of forgetting into relevance evaluation metrics could offer a more complex yet accurate representation of a user's knowledge, ultimately improving the personalization of the results.

As for the behavioural factors, previous research has studied behavioral features such as eye movements or mouse clicks that can indicate which parts of the page the user spent more time on or found most relevant. This information can be valuable in accurately representing the acquired knowledge. For example, the user may spend more time on certain sections of the page, indicating their knowledge absorption of the information, or maybe some difficulties in understanding some concepts. Further investigation should be conducted to analyze these behavioral features within the context of the knowledge acquisition model. The knowledge representation will no longer encompass the entirety of the text, but will instead include only the concepts that the user has actually learned.

The factors affecting the user's learning, do not encompass not only user-related cognitive and behavioral aspects but also document-related features. Among these, we find several document-specific factors that contribute to a user's information absorption. These factors include the keyword density and the level of vocabulary used in the document, as well as the structure and design of the page, and the language used in the document. Multimedia elements such as pictures and videos can also play a role in the user's absorption of information.

8.2.3 Extracting knowledge solely from texts

In our work, we assume that the user's knowledge is solely acquired from the text present on the documents. This assumption is aligned with previous works on characterizing SAL processes (Vakkari, 2016), which have largely focused on textual learning resources. However, these pages may also contain images, illustrations, and videos that have been proven to have a strong contribution to successful learning (Hoppe et al., 2018). Modeling the knowledge gained from these multimedia elements presents a challenge, as it may not be as straightforward as text-based knowledge representation. Future research could explore the impact of multimedia elements on the user's knowledge representation and the relevance evaluation measures and develop new techniques to model this type of knowledge. Additionally, considering user preferences for multimedia content could further enhance the personalization of search results. Overall, incorporating multimedia elements into the user profiling techniques and relevance evaluation measures could provide a more comprehensive and personalized representation of the user's knowledge.

8.2.4 Externally set information needs

When evaluating the user's knowledge or learning outcome, our experiments relied on externally set information needs, such as assigning a specific topic to search about. While this approach makes it easier to represent the user's target knowledge state, additional investigation is needed to define the set of information intent to be known. Automatic detection of the user's intent, as explored in existing literature, could be used to overcome the limitation of relying on externally set information needs. Moreover, while our experiments assumed a predefined target knowledge state, future work could explore the effects of self-set versus externally-set goals. Previous research has found that self-set goals provide three advantages, including higher performance due to increased difficulty (Latham, Mitchell, & Dossett, 1978). Azevedo *et al.* also argued that allowing students to set their own learning goals can enhance their commitment to attaining them, which is necessary for goals to affect performance (Azevedo, Ragan, Cromley, & Pritchett, 2002). Thus, future work could explore the effectiveness of our framework when the user has their own defined goals. Some goal and intention detection methods could be integrated to first classify the type of task the user is trying to achieve (Marchionini, 2006), and then detect the specific task itself (R. Yu, Dietze, et al., 2022).

Bibliography

Abualsaud, M. (2017). *Learning factors and determining document-level satisfaction in search-as-learning* (Mémoire de Master non publié). University of Waterloo.

Agichtein, E., Brill, E., & Dumais, S. (2006). Improving web search ranking by incorporating user behavior information. In *Proceedings of the 29th annual international acm sigir conference on research and development in information retrieval* (pp. 19–26).

Agosti, M., Fuhr, N., Toms, E., & Vakkari, P. (2014). Evaluation methodologies in information retrieval dagstuhl seminar 13441. In *Acm sigir forum* (Vol. 48, pp. 36–41).

Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In *Proceedings of the second acm international conference on web search and data mining* (pp. 5–14).

Alchourrón, C. E., Gärdenfors, P., & Makinson, D. (1985). On the logic of theory change : Partial meet contraction and revision functions. *The journal of symbolic logic*, *50*(2), 510–530.

Alechina, N., Bordini, R. H., Hübner, J. F., Jago, M., & Logan, B. (2006). Belief revision for agentspeak agents. In *AAMAS* (pp. 1288–1290). ACM.

Alechina, N., Jago, M., & Logan, B. (2005). Resource-bounded belief revision and contraction. In *International workshop on declarative agent languages and technologies* (pp. 141–154).

Allan, J. (2005). *Hard track overview in trec 2003 high accuracy retrieval from documents* (Rapport technique). MASSACHUSETTS UNIV AMHERST CENTER FOR INTELLIGENT INFORMATION RETRIEVAL.

Allan, J., Aslam, J., Belkin, N., Buckley, C., Callan, J., Croft, B., ... others (2003). Challenges in information retrieval and language modeling : report of a workshop held at the center for intelligent information retrieval, university of massachusetts amherst, september 2002. In *Acm sigir forum* (Vol. 37, pp. 31–47).

Allan, J., Croft, B., Moffat, A., & Sanderson, M. (2012). Frontiers, challenges, and opportunities for information retrieval : Report from swirl 2012 the second strategic workshop on information retrieval in lorne. In *Acm sigir forum* (Vol. 46, pp. 2–32).

Allen, B. (1991). Topic knowledge and online catalog search formulation. *The Library Quarterly*, *61*(2), 188–213.

Allen, B. (1998). Designing information systems for user abilities and tasks : An experimental study. *Online and CD-Rom Review*.

Allen, B. (2000). Individual differences and the conundrums of user-centered design : Two experiments. *Journal of the american society for information science*, *51*(6), 508–520.

Asnicar, F. A., & Tasso, C. (1997). ifweb : a prototype of user model-based intelligent agent for document filtering and navigation in the world wide web. In *Sixth international conference on user modeling* (pp. 2–5).

Azevedo, R., Ragan, S., Cromley, J. G., & Pritchett, S. (2002). Do different goal-setting conditions facilitate students' ability to regulate their learning of complex science topics with river-web ?.

Baddeley, A. (2007). *Working memory, thought, and action* (Vol. 45). OuP Oxford.

Baeza-Yates, R., Calderón-Benavides, L., & González-Caro, C. (2006). The intention behind web queries. In *String processing and information retrieval : 13th international conference, spire 2006, glasgow, uk, october 11-13, 2006. proceedings 13* (pp. 98–109).

Baeza-Yates, R., Ribeiro-Neto, B., et al. (1999). *Modern information retrieval* (Vol. 463). ACM press New York.

Bailey, P., Chen, L., Grosenick, S., Jiang, L., Li, Y., Reinholdtsen, P., ... Wong, S. (2012). User task understanding : a web search engine perspective. In *Nii shonan meeting on whole-session evaluation of interactive information retrieval systems, kanagawa, japan.*

Bailey, P., Moffat, A., Scholer, F., & Thomas, P. (2015). User variability and ir system evaluation. In *Proceedings of the 38th international acm sigir conference on research and development in information retrieval* (p. 625–634). New York, NY, USA : Association for Computing Machinery. Consulté sur https://doi.org/10.1145/2766462.2767728 doi: 10.1145/2766462.2767728

Bakos, J. Y. (1997). Reducing buyer search costs : Implications for electronic marketplaces. *Management science*, *43*(12), 1676–1692.

Balog, K., Azzopardi, L., & de Rijke, M. (2009). A language modeling framework for expert finding. *Information Processing & Management*, *45*(1), 1–19.

Balog, K., & Kenter, T. (2019). Personal knowledge graphs : A research agenda. In *Proceedings of the 2019 acm sigir international conference on theory of information retrieval* (pp. 217–220).

Begg, I. M., Gnocato, J., & Moore, W. E. (1993). A prototype intelligent user interface for real-time supervisory control systems. In *Proceedings of the 1st international conference on intelligent user interfaces* (pp. 211–214).

Belkin, N. J., & Croft, W. B. (1992). Information filtering and information retrieval : Two sides of the same coin ? *Communications of the ACM*, *35*(12), 29–38.

Belkin, N. J., Oddy, R. N., & Brooks, H. M. (1982). Ask for information retrieval : Part i. background and theory. *J. Documentation*, *38*(2), 61–71.

Bharat, K., & Mihaila, G. A. (2001). When experts agree : using non-affiliated experts to rank popular topics. In *Proceedings of the 10th international conference on world wide web* (pp. 597–602).

Bhattacharya, N., & Gwizdka, J. (2019). Measuring learning during search : Differences in interactions, eye-gaze, and semantic similarity to expert knowledge. In *Proceedings of the 2019 conference on human information interaction and retrieval* (pp. 63–71).

Biancalana, C., Micarelli, A., & Squarcella, C. (2008). Nereau : a social approach to query expansion. In *Proceedings of the 10th acm workshop on web information and data management* (pp. 95–102).

Biddix, J., Chung, C., & Park, H. (2011). Convenience or credibility ? a study of college student online research behaviors. *The Internet & Higher Education*, *14*(3), 175–182.

Bilal, D. (2000). Children's use of the yahooligans ! web search engine : I. cognitive, physical, and affective behaviors on fact-based search tasks. *Journal of the American Society for information Science*, *51*(7), 646–665.

Bilenko, M., & White, R. W. (2008). Mining the search trails of surfing crowds : identifying relevant websites from user activity. In *Proceedings of the 17th international conference on world wide web* (pp. 51–60).

Bilenko, M., White, R. W., Richardson, M., & Murray, G. C. (2008). Talking the talk vs. walking the walk : salience of information needs in querying vs. browsing. In *Proceedings of the 31st annual international acm sigir conference on research and development in information retrieval* (pp. 705–706).

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with python : analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Bloom, B. S. (1956). Taxonomy of educational objectives : The classification of educational goals. *Cognitive domain*.

Bloom, B. S., Engelhart, M. D., Furst, E., Hill, W. H., & Krathwohl, D. R. (1956). Handbook i : cognitive domain. *New York : David McKay*.

Bogers, T., & Van den Bosch, A. (2007). Comparing and evaluating information retrieval algorithms for news recommendation. In *Proceedings of the 2007 acm conference on recommender systems* (pp. 141–144).

Boole, G. (1847). *The mathematical analysis of logic*. Philosophical Library.

Bordini, R. H., Hübner, J. F., & Wooldridge, M. (2007). *Programming multi-agent systems in agentspeak using jason (wiley series in agent technology)*. Hoboken, NJ, USA : John Wiley & Sons, Inc.

Borlund, P. (2003a). The concept of relevance in ir. *J. Am. Soc. Inf. Sci. Technol.*, *54*(10), 913–925.

Borlund, P. (2003b). The concept of relevance in ir. *Journal of the American Society for information Science and Technology*, *54*(10), 913–925.

Bouidghaghen, O., Tamine-Lechani, L., Pasi, G., Cabanac, G., Boughanem, M., & da Costa Pereira, C. (2011). Prioritized aggregation of multiple context dimensions in mobile ir. In *Information retrieval technology : 7th asia information retrieval societies conference, airs 2011, dubai, united arab emirates, december 18-20, 2011. proceedings 7* (pp. 169–180).

Boyce, B. (1982). Beyond topicality : A two stage view of relevance and the retrieval process. *Information Processing & Management*, *18*(3), 105–109.

Brand-Gruwel, S., Wopereis, I., & Walraven, A. (2009). A descriptive model of information problem solving while using internet. *Computers & Education*, *53*(4), 1207–1217.

Broder, A. (2002). A taxonomy of web search. In *Acm sigir forum* (Vol. 36, pp. 3–10).

Brookes, B. C. (1980). The foundations of information science. part i. philosophical aspects. *Journal of information science*, *2*(3-4), 125–133.

Buttcher, S., Clarke, C. L., & Cormack, G. V. (2016). *Information retrieval : Implementing and evaluating search engines*. Mit Press.

Byström, K., & Järvelin, K. (1995). Task complexity affects information seeking and use. *Inf. Process. Manag.*, *31*(2), 191–213.

Cai, F., Liang, S., & De Rijke, M. (2014). Personalized document re-ranking based on bayesian probabilistic matrix factorization. In *Proceedings of the 37th international acm sigir conference on research & development in information retrieval* (pp. 835–838).

Câmara, A., El-Zein, D., & da Costa-Pereira, C. (2022). Rulk : A framework for representing user knowledge in search-as-learning.

Câmara, A., Maxwell, D., & Hauff, C. (2022a). Searching, learning, and subtopic ordering : A simulation-based analysis. In *Advances in information retrieval. ecir 2022. lecture notes in computer science, vol 13185* (pp. 142–156).

Câmara, A., Maxwell, D., & Hauff, C. (2022b). Searching, learning, and subtopic ordering : A simulation-based analysis. In *Ecir* (pp. 142–156).

Câmara, A., Roy, N., Maxwell, D., & Hauff, C. (2021a). Searching to learn with instructional scaffolding. In *Proceedings of the 2021 conference on human information interaction and retrieval* (pp. 209–218).

Câmara, A., Roy, N., Maxwell, D., & Hauff, C. (2021b). Searching to learn with instructional scaffolding. *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*.

Campos, R., Mangaravite, V., Pasquali, A., Jorge, A. M., Nunes, C., & Jatowt, A. (2018). Yake ! collection-independent automatic keyword extractor. In *European conference on information retrieval* (pp. 806–810).

Carrillo-Ramos, A., Gensel, J., Villanova-Oliver, M., & Martin, H. (2005). Pumas : a framework based on ubiquitous agents for accessing web information systems through mobile devices. In *Proceedings of the 2005 acm symposium on applied computing* (pp. 1003–1008).

Castro, J., Rodriguez, R. M., & Barranco, M. J. (2014). Weighting of features in content-based filtering with entropy and dependence measures. *International journal of computational intelligence systems*, *7*(1), 80–89.

Chen, H., & Karger, D. R. (2006). Less is more : probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th annual international acm sigir conference on research and development in information retrieval* (pp. 429–436).

Chen, L., & Sycara, K. (1998). Webmate : A personal agent for browsing and searching. In *Proceedings of the second international conference on autonomous agents* (pp. 132–139).

Chi, Y., Han, S., He, D., & Meng, R. (2016). Exploring knowledge learning in collaborative information seeking process. In *Ceur workshop proceedings* (Vol. 1647).

Chiir '16 : Proceedings of the 2016 acm on conference on human information interaction and retrieval. (2016). New York, NY, USA : Association for Computing Machinery.

Chirita, P.-A., Olmedilla, D., & Nejdl, W. (2004). Pros : A personalized ranking platform for web search. In *International conference on adaptive hypermedia and adaptive web-based systems* (pp. 34–43).

Clarke, C. L., Kolla, M., Cormack, G. V., Vechtomova, O., Ashkan, A., Büttcher, S., & MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international acm sigir conference on research and development in information retrieval* (pp. 659–666).

Cleverdon, C. (1967). The cranfield tests on index language devices. In *Aslib proceedings* (Vol. 19, pp. 173–194).

Coiro, J. (2011). Predicting reading comprehension on the internet : Contributions of offline reading skills, online reading skills, and prior knowledge. *Journal of literacy research*, *43*(4), 352–392.

Colbert-Getz, J. M., Fleishman, C., Jung, J., & Shilkofski, N. (2013). How do gender and anxiety affect students' self-assessment and actual performance on a high-stakes clinical skills examination ? *Academic medicine*, *88*(1), 44–48.

Cole, M. J., Gwizdka, J., Liu, C., Belkin, N. J., & Zhang, X. (2013). Inferring user knowledge level from eye movement patterns. *Information Processing and Management*, *49*(5), 1075–1091.

Collins-Thompson, K., & Callan, J. P. (2004). A language modeling approach to predicting reading difficulty. In *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics : Hlt-naacl 2004* (pp. 193–200).

Collins-Thompson, K., Hansen, P., & Hauff, C. (2017). Search as learning (dagstuhl seminar 17092). In *Dagstuhl reports* (Vol. 7).

Collins-Thompson, K., Rieh, S. Y., Haynes, C. C., & Syed, R. (2016). Assessing learning outcomes in web search : A comparison of tasks and query strategies. In *Proceedings of the 2016 acm on conference on human information interaction and retrieval* (pp. 163–172).

Constantino, G. D., & Raffaghelli, J. E. (2020). Online teaching and learning : going beyond the information given. *Cultural Views on Online Learning in Higher Education : A Seemingly Borderless Class*, 3–28.

Cool, C., & Spink, A. (2002). *Issues of context in information retrieval (ir) : an introduction to the special issue* (Vol. 38) (N° 5). Elsevier.

Corbett, A. T., & Anderson, J. R. (1994). Knowledge tracing : Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, *4*, 253–278.

Craswell, N., Mitra, B., Yilmaz, E., Campos, D., & Lin, J. (2021). MS MARCO : benchmarking ranking models in the large-data regime. In *Sigir* (pp. 1566–1576). Acm.

Croft, W. B., Metzler, D., & Strohman, T. (2010). *Search engines : Information retrieval in practice* (Vol. 520). Addison-Wesley Reading.

Culpepper, J. S., Diaz, F., & Smucker, M. D. (2018). Research frontiers in information retrieval : Report from the third strategic workshop on information retrieval in lorne (swirl 2018). In *Acm sigir forum* (Vol. 52, pp. 34–90).

Cutrell, E., & Guan, Z. (2007). What are you looking for ? an eye-tracking study of information usage in web search. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 407–416).

da Costa Móra, M., Lopes, J. G. P., Vicari, R. M., & Coelho, H. (1998). Bdi models and systems : Bridging the gap. In *Atal* (pp. 11–27).

da Costa Pereira, C., Dragoni, M., & Pasi, G. (2009). Multidimensional relevance : A new aggregation criterion. In *Advances in information retrieval : 31th european conference on ir research, ecir 2009, toulouse, france, april 6-9, 2009. proceedings 31* (pp. 264–275).

Daoud, M., Lechani, L.-T., & Boughanem, M. (2009). Towards a graph-based user profile modeling for a session-based personalized search. *Knowledge and Information Systems*, *21*(3), 365–398.

Daoud, M., Tamine, L., & Boughanem, M. (2010). A personalized graph-based document ranking model using a semantic user profile. In *International conference on user modeling, adaptation, and personalization* (pp. 171–182).

Davies, S., Butcher, K., & Stevens, C. (2013). Self-regulated learning with graphical overviews : When spatial information detracts from learning. In *Proceedings of the annual meeting of the cognitive science society* (Vol. 35).

de Campos, L. M., Fernández-Luna, J. M., Huete, J. F., & Vicente-López, E. (2013). Using personalization to improve xml retrieval. *IEEE Transactions on Knowledge and Data Engineering*, *26*(5), 1280–1292.

Demaree, D., Jarodzka, H., Brand-Gruwel, S., & Kammerer, Y. (2020). The influence of device type on querying behavior and learning outcomes in a searching as learning task with a laptop or smartphone. In *Proceedings of the 2020 conference on human information interaction and retrieval* (pp. 373–377).

Demeester, T., Trieschnigg, D., Nguyen, D., Zhou, K., & Hiemstra, D. (2014). *Overview of the trec 2014 federated web search track* (Rapport technique). GHENT UNIV (BELGIUM).

De Rosa, C., Cantrell, J., Hawk, J., & Wilson, A. (2006). *College students' perceptions of libraries and information resources : A report to the oclc membership*. Online Computer Library Center,.

DeStefano, D., & LeFevre, J.-A. (2007). Cognitive load in hypertext reading : A review. *Computers in human behavior*, *23*(3), 1616–1641.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert : Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, *abs/1810.04805*.

Dietz, L., Gamari, B., Dalton, J., & Craswell, N. (2018). TREC complex answer retrieval overview. In *Trec* (Vol. 500-331). National Institute of Standards and Technology (NIST).

Diffbot. (s. d.). *Diffbot natural language api demo.* https://www.diffbot.com/products/natural-language/. (Accessed on 19 April 2023)

Ding, C., & Patra, J. C. (2007). User modeling for personalized web search with self-organizing map. *Journal of the American Society for information Science and Technology*, *58*(4), 494–507.

Dubois, D., Lang, J., & Prade, H. (1991). A possibilistic assumption-based truth maintenance system with uncertain justifications, and its application to belief revision. In *Truth maintenance systems : Ecai-90 workshop stockholm, sweden, august 6, 1990 proceedings* (pp. 87–106).

Duggan, G. B., & Payne, S. J. (2008). Knowledge in the head and on the web : Using topic expertise to aid search. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 39–48).

Dutton, W. H., & Helsper, E. J. (2007). Oxford internet survey 2007 report : The internet in britain. *Available at SSRN 1327033*.

Eickhoff, C., Collins-Thompson, K., Bennett, P. N., & Dumais, S. (2013). Personalizing atypical web search sessions. In *Proceedings of the sixth acm international conference on web search and data mining* (pp. 285–294).

Eickhoff, C., Gwizdka, J., Hauff, C., & He, J. (2017). Introduction to the special issue on search as learning. *Information Retrieval Journal*, *20*, 399–402.

Ekstrom, R. B., French, J. W., Harman, H. H., & Dermen, D. (1976). *Manual for kit of factor-referenced cognitive tests : 1976*.

El Zein, D. (2021). User knowledge and search goals in information retrieval : A benchmark and study on the evolution of users' knowledge gain. In *Desires* (pp. 189–190).

El Zein, D. (2022). Cognitive information retrieval. In *Advances in information retrieval : 44th european conference on ir research, ecir 2022, stavanger, norway, april 10–14, 2022, proceedings, part ii* (pp. 473–479).

El Zein, D., Câmara, A., Da Costa Pereira, C., & Tettamanzi, A. (2023). Rulkne : Representing user knowledge state in search-as-learning with named entities. In *Proceedings of the 2023 conference on human information interaction and retrieval* (pp. 388–393).

El Zein, D., & da Costa Pereira, C. (2020a). A cognitive agent framework in information retrieval : Using user beliefs to customize results. In *International conference on principles and practice of multi-agent systems* (pp. 325–333).

El Zein, D., & da Costa Pereira, C. (2020b). Graded belief revision for jason : A rule-based approach. In *2020 ieee/wic/acm international joint conference on web intelligence and intelligent agent technology (wi-iat)* (pp. 211–218).

El Zein, D., & da Costa Pereira, C. (2021a). A cognitive agent framework in information retrieval : Using user beliefs to customize results. In *Prima 2020 : Principles and practice of multi-agent systems : 23rd international conference, nagoya, japan, november 18–20, 2020, proceedings 23* (pp. 325–333).

El Zein, D., & da Costa Pereira, C. (2021b). Representing, tracking and revising the user's knowledge : A search result filter framework. In *Iir*.

El Zein, D., & da Costa Pereira, C. (2022a). Jason agents for knowledge-aware information retrieval filters. In *Icaart (2)* (pp. 466–476).

El Zein, D., & da Costa Pereira, C. (2022b). User's knowledge and information needs in information retrieval evaluation. In *Proceedings of the 30th acm conference on user modeling, adaptation and personalization* (pp. 170–178).

El Zein, D., & Da Costa Pereira, C. (2023). The evolution of user knowledge during search-as-learning sessions : A benchmark and baseline. In *Proceedings of the 2023 conference on human information interaction and retrieval* (pp. 454–458).

El Zein, D., & Pereira, C. (2022). Accounting for user's knowledge and search goals in information retrieval evaluation. In *Proceedings of the 2nd joint conference of the information retrieval communities in europe (circle 2022), samatan, gers, france, july 4-7, 2022*.

Freund, L., He, J., Gwizdka, J., Kando, N., Hansen, P., & Rieh, S. Y. (2014). Searching as learning (sal) workshop 2014. In *Proceedings of the 5th information interaction in context symposium* (pp. 7–7).

Freund, L., Kopak, R., & O'Brien, H. (2016). The effects of textual environment on reading comprehension : Implications for searching as learning. *Journal of Information Science*, *42*(1), 79–93.

Frias-Martinez, E., Chen, S. Y., Macredie, R. D., & Liu, X. (2007). The role of human factors in stereotyping behavior and perception of digital library users : a robust clustering approach.

Furnas, G. W., Landauer, T. K., Gomez, L. M., & Dumais, S. T. (1987). The vocabulary problem in human-system communication. *Communications of the ACM*, *30*(11), 964–971.

Gadiraju, U., Yu, R., Dietze, S., & Holtz, P. (2018). Analyzing knowledge gain of users in informational search sessions on the web. In *Proceedings of the 2018 conference on human information interaction & retrieval* (pp. 2–11).

Gärdenfors, P., & Makinson, D. (1988). Revisions of knowledge systems using epistemic entrenchment. In *Proceedings of the 2nd conference on theoretical aspects of reasoning about knowledge* (pp. 83–95).

Gauch, S., Speretta, M., Chandramouli, A., & Micarelli, A. (2007). User profiles for personalized information access. *The adaptive web*, 54–89.

Ge, S., Dou, Z., Jiang, Z., Nie, J.-Y., & Wen, J.-R. (2018). Personalizing search results using hierarchical rnn with query-aware attention. In *Proceedings of the 27th acm international conference on information and knowledge management* (pp. 347–356).

Ghorab, M. R., Zhou, D., O'connor, A., & Wade, V. (2013). Personalised information retrieval : survey and classification. *User Modeling and User-Adapted Interaction*, *23*(4), 381–443.

Ghosh, S., Rath, M., & Shah, C. (2018). Searching as learning : Exploring search behavior and learning outcomes in learning-related tasks. In *Proceedings of the 2018 conference on human information interaction & retrieval* (pp. 22–31).

Goffman, W. (1964). A searching procedure for information retrieval. *Information Storage and Retrieval*, *2*(2), 73–78.

González-Betancor, S. M., Bolívar-Cruz, A., & Verano-Tacoronte, D. (2019). Self-assessment accuracy in higher education : The influence of gender and performance of university students. *Active learning in higher education*, *20*(2), 101–114.

Griffiths, J. R., & Brophy, P. (2005). Student searching behavior and the web : use of academic resources and google.

Guttman, R. H., & Maes, P. (1998). Agent-mediated integrative negotiation for retail electronic commerce. In *International workshop on agent-mediated electronic trading* (pp. 70–90).

Gwizdka, J. (2010). Distribution of cognitive load in web search. *Journal of the American Society for Information Science and Technology*, *61*(11), 2167–2187.

Gwizdka, J. (2017). I can and so i search more : effects of memory span on search behavior. In *Proceedings of the 2017 conference on conference human information interaction and retrieval* (pp. 341–344).

Gwizdka, J., & Chen, X. (2016). Towards observable indicators of learning on search. In *Sal@ sigir*.

Gwizdka, J., Hansen, P., Hauff, C., He, J., & Kando, N. (2016). Search as learning (sal) workshop 2016. In *Proceedings of the 39th international acm sigir conference on research and development in information retrieval* (pp. 1249–1250).

Hagen, M., Gomoll, J., Beyer, A., & Stein, B. (2013). From search session detection to search mission detection. In *Proceedings of the 10th conference on open research areas in information retrieval* (pp. 85–92).

Hansen, P., & Rieh, S. Y. (2016). Recent advances on searching as learning : An introduction to the special issue. *Journal of Information Science*, *42*(1), 3–6.

Harman, D. K. (1996). Overview of the fourth text retrieval conference (trec-4).

Harter, S. P., & Hert, C. A. (1997). Evaluation of information retrieval systems : Approaches, issues, and methods. *Annual Review of Information Science and Technology (ARIST)*, *32*, 3–94.

Hassan Awadallah, A., White, R., Pantel, P., Dumais, S., & Wang, Y.-M. (2014). Supporting complex search tasks. In *Proc. 23rd acm cikm* (pp. 829–838).

Hembrooke, H. A., Granka, L. A., Gay, G. K., & Liddy, E. D. (2005). The effects of expertise and feedback on search term selection and subsequent learning. *Journal of the American Society for Information Science and Technology*, *56*(8), 861–871.

Hingoro, M. A., & Nawaz, H. (2021). A comparative analysis of search engine ranking algorithms. *International Journal*, *10*(2).

Hölscher, C., & Strube, G. (2000). Web search behavior of internet experts and newbies. *Computer networks*, *33*(1-6), 337–346.

Honnibal, M., & Montani, I. (2017). *spaCy 2 : Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.* (To appear)

Hoppe, A., Holtz, P., Kammerer, Y., Yu, R., Dietze, S., & Ewerth, R. (2018). Current challenges for studying search as learning processes. In *Linked learning workshop – learning and education with web data (lile), in conjunction with acm conference on web science.*

Hsieh-Yee, I. (1993). Effects of search experience and subject knowledge on the search tactics of novice and experienced searchers. *Journal of the american society for information science*, *44*(3), 161–174.

Hu, J., Wang, G., Lochovsky, F., Sun, J.-t., & Chen, Z. (2009). Understanding user's query intent with wikipedia. In *Proceedings of the 18th international conference on world wide web* (pp. 471–480).

Huang, X., & Soergel, D. (2013). Relevance : An improved framework for explicating the notion. *Journal of the American Society for Information Science and Technology*, *64*(1), 18–35.

Hubert, G., & Cabanac, G. (2012). *Irit at trec 2012 contextual suggestion track* (Rapport technique). TOULOUSE UNIV (FRANCE).

Hupfer, M. E., & Detlor, B. (2006). Gender and web information seeking : A self-concept orientation model. *Journal of the American Society for Information Science and Technology*, *57*(8), 1105–1115.

Iiix : Proceedings of the 1st international conference on information interaction in context. (2006). New York, NY, USA : Association for Computing Machinery.

Ilkou, E. (2022). Personal knowledge graphs : Use cases in e-learning platforms. In *Companion proceedings of the web conference 2022* (pp. 344–348).

Ingwersen, P. (1992). *Information retrieval interaction* (Vol. 246). Taylor Graham London.

Ingwersen, P., & Järvelin, K. (2005). Information retrieval in context : Irix. In *Acm sigir forum* (Vol. 39, pp. 31–39).

Jansen, B. J., Booth, D. L., & Spink, A. (2007). Determining the user intent of web search engine queries. In *Proceedings of the 16th international conference on world wide web* (pp. 1149–1150).

Jansen, B. J., Booth, D. L., & Spink, A. (2008). Determining the informational, navigational, and transactional intent of web queries. *Information Processing & Management*, *44*(3), 1251–1266.

Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, *20*(4), 422–446.

Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, *20*(4), 422–446.

Järvelin, K., & Kekäläinen, J. (2017). Ir evaluation methods for retrieving highly relevant documents. In *Acm sigir forum* (Vol. 51, pp. 243–250).

Jason agent programming. (2021). http://jason.sourceforge.net/wp/.

Jensen, A. S., & Villadsen, J. (2015). Plan-belief revision in jason. In *ICAART (1)* (pp. 182–189). SciTePress.

Jiang, J., He, D., & Allan, J. (2014). Searching, browsing, and clicking in a search session : changes in user behavior by task and over time. In *Proceedings of the 37th international acm sigir conference on research & development in information retrieval* (pp. 607–616).

Judd, T., & Kennedy, G. (2010). A five-year study of on-campus internet use by undergraduate biomedical students. *Computers & Education*, *55*(4), 1564–1571.

Kalyani, R., & Gadiraju, U. (2019). Understanding user search behavior across varying cognitive levels. In *Proceedings of the 30th acm conference on hypertext and social media* (pp. 123–132).

Kang, R., & Fu, W.-T. (2010). Exploratory information search by domain experts and novices. In *Proceedings of the 15th international conference on intelligent user interfaces* (pp. 329–332).

Kathuria, A., Jansen, B. J., Hafernik, C., & Spink, A. (2010). Classifying the user intent of web queries using k-means clustering. *Internet Research*, *20*(5), 563–581.

Kelly, D., & Cool, C. (2002). The effects of topic familiarity on information search behavior. In *Proceedings of the 2nd acm/ieee-cs joint conference on digital libraries* (pp. 74–75).

Kelly, D., Kantor, P., Morse, E., Scholtz, J., & Sun, Y. (2006). User-centered evaluation of interactive question answering systems. In *Proceedings of the interactive question answering workshop at hlt-naacl 2006* (pp. 49–56).

Kelly, D., et al. (2009). Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends® in Information Retrieval*, *3*(1–2), 1–224.

Kim, H. R., & Chan, P. K. (2003). Learning implicit user interest hierarchy for context in personalization. In *Proceedings of the 8th international conference on intelligent user interfaces* (pp. 101–108).

Kim, K.-S. (2008). Effects of emotion control and task on web searching behavior. *Information Processing & Management*, *44*(1), 373–385.

Kim, K.-S., & Allen, B. (2002). Cognitive and task influences on web searching behavior. *Journal of the American Society for Information Science and Technology*, *53*(2), 109–119.

Kline, T. J. (2005). *Psychological testing : A practical approach to design and evaluation*. Sage publications.

Koutrika, G., & Ioannidis, Y. (2005). A unified user profile framework for query disambiguation and personalization. In *Proceedings of the workshop on new technologies for personalized information access (pia2005), edinburgh, scotland, uk* (pp. 44–53).

Krathwohl, D. R. (2002). A revision of bloom's taxonomy : An overview. *Theory into practice*, *41*(4), 212–218.

Kravi, E., Guy, I., Mejer, A., Carmel, D., Maarek, Y., Pelleg, D., & Tsur, G. (2016). One query, many clicks : Analysis of queries with multiple clicks by the same user. In *Proceedings of the 25th acm international on conference on information and knowledge management* (pp. 1423–1432).

Kurumatani, K. (2004). Multi-agent for mass user support. *Lecture Notes in Artificial Intelligence (LNAI)*, *3012*.

Lafferty, J., & Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international acm sigir conference on research and development in information retrieval* (pp. 111–119).

Latham, G. P., Mitchell, T. R., & Dossett, D. L. (1978). Importance of participative goal setting and anticipated rewards on goal difficulty and job performance. *Journal of applied psychology*, *63*(2), 163.

Lee, U., Liu, Z., & Cho, J. (2005). Automatic identification of user goals in web search. In *Proceedings of the 14th international conference on world wide web* (pp. 391–400).

Li, Y., & Belkin, N. J. (2008). A faceted approach to conceptualizing tasks in information seeking. *Information processing & management*, *44*(6), 1822–1837.

Lieberman, H. (1997). Autonomous interface agents. In *Proceedings of the acm sigchi conference on human factors in computing systems* (pp. 67–74).

Lin, J., Nogueira, R., & Yates, A. (2020). Pretrained transformers for text ranking : BERT and beyond. *CoRR*, *abs/2010.06467*.

Liu, H., & Hoeber, O. (2011). A luhn-inspired vector re-weighting approach for improving personalized web search. In *2011 ieee/wic/acm international conferences on web intelligence and intelligent agent technology* (Vol. 3, pp. 301–305).

Liu, H., Liu, C., & Belkin, N. J. (2019). Investigation of users' knowledge change process in learning-related search tasks. *Proceedings of the Association for Information Science and Technology*, *56*(1), 166–175.

Liu, J., & Belkin, N. J. (2010). Personalizing information retrieval for people with different levels of topic knowledge. In *Proceedings of the 10th annual joint conference on digital libraries* (pp. 383–384).

Liu, J., Belkin, N. J., Zhang, X., & Yuan, X. (2013). Examining users' knowledge change in the task completion process. *Information Processing & Management*, *49*(5), 1058–1074.

Liu, J., & Jung, Y. J. (2021). Interest development, knowledge learning, and interactive ir : toward a state-based approach to search as learning. In *Proceedings of the 2021 conference on human information interaction and retrieval* (pp. 239–248).

Liu, J., Liu, C., & Belkin, N. J. (2016a). Predicting information searchers' topic knowledge at different search stages. *J. Assoc. Inf. Sci. Technol.*, *67*(11), 2652–2666.

Liu, J., Liu, C., & Belkin, N. J. (2016b). Predicting information searchers' topic knowledge at different search stages. *Journal of the Association for Information Science and Technology*, *67*(11), 2652–2666.

Liu, J., Liu, C., & Belkin, N. J. (2020). Personalization in text information retrieval : A survey. *Journal of the Association for Information Science and Technology*, *71*(3), 349–369.

Liu, S., Yu, C., & Meng, W. (2005). Word sense disambiguation in queries. In *Proceedings of the 14th acm international conference on information and knowledge management* (pp. 525–532).

Liu, Y., Zhang, M., Ru, L., & Ma, S. (2006). Automatic query type identification based on click through information. In *Information retrieval technology : Third asia information retrieval symposium, airs 2006, singapore, october 16-18, 2006. proceedings 3* (pp. 593–600).

Lu, S., Dou, Z., Jun, X., Nie, J.-Y., & Wen, J.-R. (2019). Psgan : A minimax game for personalized search with limited and noisy click data. In *Proceedings of the 42nd international acm sigir conference on research and development in information retrieval* (pp. 555–564).

MacFarlane, A., Al-Wabil, A., Marshall, C. R., Albrair, A., Jones, S. A., & Zaphiris, P. (2010). The effect of dyslexia on information retrieval : A pilot study. *Journal of Documentation*, *66*(3), 307–326.

Machado, M. d. O. C., de Alcantara Gimenez, P. J., & Siqueira, S. W. M. (2020). Raising the dimensions and variables for searching as a learning process : a systematic mapping of the literature. In *Anais do xxxi simpósio brasileiro de informática na educação* (pp. 1393–1402).

Madhusudan, P. A., & Poonam, D. L. (2017). Deep web crawling efficiently using dynamic focused web crawler. *InternationalReG searchJournalofEngineeringand Technology*, *4*(6), 3303G3306.

Malabonga, V., Kenyon, D. M., & Carpenter, H. (2005). Self-assessment, preparation and response time on a computerized oral proficiency test. *Language Testing*, *22*(1), 59–92.

Manning, C. D. (2008). *Introduction to information retrieval*. Syngress Publishing,.

Mao, J., Liu, Y., Luan, H., Zhang, M., Ma, S., Luo, H., & Zhang, Y. (2017). Understanding and predicting usefulness judgment in web search. In *Proceedings of the 40th international acm sigir conference on research and development in information retrieval* (pp. 1169–1172).

Mao, J., Liu, Y., Zhou, K., Nie, J.-Y., Song, J., Zhang, M., … Luo, H. (2016). When does relevance mean usefulness and user satisfaction in web search ? In *Proceedings of the 39th international acm sigir conference on research and development in information retrieval* (pp. 463–472).

Marchionini, G. (1993). Information seeking in full-text end-user-oriented search systems : The roles of domain and search expertise. *Library & information science research*, *15*(1), 35–69.

Marchionini, G. (1995). *Information seeking in electronic environments*. Cambridge University Press. doi: 10.1017/CBO9780511626388

Marchionini, G. (2006). Exploratory search : from finding to understanding. *Comm. ACM*, *49*(4), 41–46.

Marchionini, G., & Maurer, H. (1995). The roles of digital libraries in teaching and learning. *Communications of the ACM*, *38*(4), 67–75.

Matthijs, N., & Radlinski, F. (2011). Personalizing web search using long term browsing history. In *Proceedings of the fourth acm international conference on web search and data mining* (pp. 25–34).

Maxwell, D., & Azzopardi, L. (2018). Information scent, searching and stopping : Modelling serp level stopping behaviour. In *Advances in information retrieval : 40th european conference on ir research, ecir 2018, grenoble, france, march 26-29, 2018, proceedings 40* (pp. 210–222).

Mayer, R. E. (1997). Multimedia learning : Are we asking the right questions ? *Educational psychologist*, *32*(1), 1–19.

Mc Gowan, J. P. (2003). *A multiple model approach to personalised information access* (Thèse de doctorat non publiée). Citeseer.

Mendes, P. N., Jakob, M., García-Silva, A., & Bizer, C. (2011). Dbpedia spotlight : shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems* (pp. 1–8).

Micarelli, A., & Sciarrone, F. (2004). Anatomy and empirical evaluation of an adaptive web-based information filtering system. *User Modeling and User-Adapted Interaction*, *14*(2), 159–200.

Mitrovic, A., & Martin, B. (2007). Evaluating the effect of open student models on self-assessment. *International Journal of Artificial Intelligence in Education*, *17*(2), 121–144.

Moffat, A., Thomas, P., & Scholer, F. (2013). Users versus models : What observation tells us about effectiveness metrics. In *Proceedings of the 22nd acm international conference on information & knowledge management* (pp. 659–668).

Monchaux, S., Amadieu, F., Chevalier, A., & Mariné, C. (2015). Query strategies during information searching : Effects of prior domain knowledge and complexity of the information problems to be solved. *Information Processing & Management*, *51*(5), 557–569.

Mostafa, J., Mukhopadhyay, S., & Palakal, M. (2003). Simulation studies of different dimensions of users' interests and their impact on user modeling and information filtering. *Information Retrieval*, *6*, 199–223.

Naumann, J., Richter, T., Christmann, U., & Groeben, N. (2008). Working memory capacity and reading skill moderate the effectiveness of strategy training in learning from hypertext. *Learning and Individual Differences*, *18*(2), 197–213.

Navigli, R., & Crisafulli, G. (2010). Inducing word senses to improve web search result clustering. In *Proceedings of the 2010 conference on empirical methods in natural language processing* (pp. 116–126).

Netday. (2004). *Netday 2004.* Turtle,Voices and views of today's tech-savvy students : National report on NetDay speak up day for students 2003. (2004).

O'Brien, H. L., Kampen, A., Cole, A. W., & Brennan, K. (2020a). The role of domain knowledge in search as learning. In *Proceedings of the 2020 conference on human information interaction and retrieval* (pp. 313–317).

O'Brien, H. L., Kampen, A., Cole, A. W., & Brennan, K. (2020b). The role of domain knowledge in search as learning. In *Chiir* (pp. 313–317). Acm.

Otto, C., Yu, R., Pardi, G., Hoyer, J. v., Rokicki, M., Hoppe, A., ... Ewerth, R. (2021). Predicting knowledge gain during web search based on multimedia resource consumption. In *International conference on artificial intelligence in education* (pp. 318–330).

Pardi, G., von Hoyer, J., Holtz, P., & Kammerer, Y. (2020). The role of cognitive abilities and time spent on texts and videos in a multimodal searching as learning task. In *Proceedings of the 2020 conference on human information interaction and retrieval* (pp. 378–382).

Park, T. K. (1994). Toward a theory of user-based relevance : A call for a new paradigm of inquiry. *Journal of the American society for information science*, *45*(3), 135–141.

Pasi, G. (2010). Issues in personalizing information retrieval. *IEEE Intell. Informatics Bull.*, *11*(1), 3–7.

Pasi, G., Bordogna, G., & Villa, R. (2007, 01). A multi-criteria content-based filtering system. In (p. 775-776).

Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. *The adaptive web : methods and strategies of web personalization*, 325–341.

Peters, C., & Ferro, N. (Eds.). (2014). *Working notes for CLEF 2000 workshop co-located with the 4th european conference on digital libraries (ECDL 2000), lisbon, portugal, september 21-22, 2000* (Vol. 1166). CEUR-WS.org. Consulté sur https://ceur-ws.org/Vol-1166

Ponte, J. M., & Croft, W. B. (2017). A language modeling approach to information retrieval. In *Acm sigir forum* (Vol. 51, pp. 202–208).

Psarras, I., & Jose, J. (2006). A system for adaptive information retrieval. In *International conference on adaptive hypermedia and adaptive web-based systems* (pp. 313–317).

Putra, S. R., Moraes, F., & Hauff, C. (2018). Searchx : Empowering collaborative search research. In *Sigir* (pp. 1265–1268). Acm.

Qiu, F., & Cho, J. (2006). Automatic identification of user interest for personalized search. In *Proceedings of the 15th international conference on world wide web* (pp. 727–736).

Qiu, S., Gadiraju, U., & Bozzon, A. (2020). Towards memorable information retrieval. In *Proceedings of the 2020 acm sigir on international conference on theory of information retrieval* (pp. 69–76).

Rao, A., & Georgeff, M. (2001, 01). Modeling rational agents within a bdi-architecture.

Reimers, N., & Gurevych, I. (2019). Sentence-bert : Sentence embeddings using siamese bert-networks. In *Emnlp/ijcnlp (1)* (pp. 3980–3990). Association for Computational Linguistics.

Robertson, S. E., & Jones, K. S. (1976). Relevance weighting of search terms. *Journal of the American Society for Information science*, *27*(3), 129–146.

Rocchio Jr, J. J. (1971). Relevance feedback in information retrieval. *The SMART retrieval system : experiments in automatic document processing*.

Rose, D. E., & Levinson, D. (2004a). Understanding user goals in web search. In *Proceedings of the 13th international conference on world wide web* (pp. 13–19).

Rose, D. E., & Levinson, D. (2004b). Understanding user goals in web search. In *Proceedings of the 13th international conference on world wide web* (pp. 13–19).

Roy, N., Moraes, F., & Hauff, C. (2020). Exploring users' learning gains within search sessions. *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval*.

Roy, N., Torre, M. V., Gadiraju, U., Maxwell, D., & Hauff, C. (2021). Note the highlight : incorporating active reading tools in a search as learning environment. In *Proceedings of the 2021 conference on human information interaction and retrieval* (pp. 229–238).

Russell, D. M., Tang, D., Kellar, M., & Jeffries, R. (2009). Task behaviors during web search : The difficulty of assigning labels. In *2009 42nd hawaii international conference on system sciences* (pp. 1–5).

Safavi, T., Belth, C., Faber, L., Mottin, D., Müller, E., & Koutra, D. (2019). Personalized knowledge graph summarization : From the cloud to your pocket. In *2019 ieee international conference on data mining (icdm)* (pp. 528–537).

Saito, H., Egusa, Y., Terai, H., Kando, N., Nakashima, R., Takaku, M., & Miwa, M. (2011). Changes in users' knowledge structures before and after web search on a topic : Analysis using the concept map. *Proceedings of the American Society for Information Science and Technology*, *48*(1), 1–4.

Salimzadeh, S., Gadiraju, U., Hauff, C., & van Deursen, A. (2022). Exploring the feasibility of crowd-powered decomposition of complex user questions in text-to-sql tasks. In *Ht* (pp. 154–165). Acm.

Salton, G., Wong, A., & Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, *18*(11), 613–620.

Salton, G., & Yang, C.-S. (1973). On the specification of term values in automatic indexing. *Journal of documentation*.

Sanchiz, M., Chevalier, A., & Amadieu, F. (2017). How do older and young adults start searching for information ? impact of age, domain knowledge and problem complexity on the different steps of information searching. *Computers in Human Behavior*, *72*, 67–78.

Sanchiz, M., Chin, J., Chevalier, A., Fu, W.-T., Amadieu, F., & He, J. (2017). Searching for information on the web : Impact of cognitive aging, prior domain knowledge and complexity of the search problems. *Information Processing & Management*, *53*(1), 281–294.

Saracevic, T. (1970). *On the concept of relevance in information science*. Case Western Reserve University.

Saracevic, T. (1975). Relevance : A review of and a framework for the thinking on the notion in information science. *Journal of the American Society for information science*, *26*(6), 321–343.

Saracevic, T., & Kantor, P. (1988). A study of information seeking and retrieving. iii. searchers, searches, and overlap. *Journal of the American Society for information Science*, *39*(3), 197–216.

Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39). Cambridge University Press Cambridge.

Selwyn, N. (2008). An investigation of differences in undergraduates' academic use of the internet. *Active learning in higher education*, *9*(1), 11–22.

Sharit, J., Hernández, M. A., Czaja, S. J., & Pirolli, P. (2008). Investigating the roles of knowledge and cognitive abilities in older adult information seeking on the web. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *15*(1), 1–25.

Sieg, A., Mobasher, B., & Burke, R. (2004). Inferring user's information context from user profiles and concept hierarchies. In *Classification, clustering, and data mining applications : Proceedings of the meeting of the international federation of classification societies (ifcs), illinois institute of technology, chicago, 15–18 july 2004* (pp. 563–573).

Sieg, A., Mobasher, B., & Burke, R. (2007). Web search personalization with ontological user profiles. In *Proceedings of the sixteenth acm conference on conference on information and knowledge management* (pp. 525–534).

Sihvonen, A., & Vakkari, P. (2004). Subject knowledge improves interactive query expansion assisted by a thesaurus. *Journal of Documentation*, *60*(6), 673–690.

Sluis, F. v. d., & Broek, E. L. (2010). Modeling user knowledge from queries : Introducing a metric for knowledge. In *International conference on active media technology* (pp. 395–402).

Soboroff, I., Huang, S., & Harman, D. (2018). Trec 2018 news track overview. In *Trec* (Vol. 409, p. 410).

Song, D., & Bruza, P. (2003, 02). Towards context sensitive information inference. *Journal of the American Society for Information Science and Technology*, *54*, 321 - 334. doi: 10.1002/asi.10213

Speretta, M., & Gauch, S. (2005a). misearch. In *The 2005 ieee/wic/acm international conference on web intelligence (wi'05)* (pp. 807–808).

Speretta, M., & Gauch, S. (2005b). Personalized search based on user search histories. In *The 2005 ieee/wic/acm international conference on web intelligence (wi'05)* (pp. 622–628).

Stahl, K. A. D., & Bravo, M. A. (2010). Contemporary classroom / vocabulary assessment / for content areas. *The Reading Teacher*, *63*, 566–578.

Stefani, A., & Strapparava, C. (1998). Determinazione automatica del profilo dell'utente web : il sistema siteif.

Syed, R., & Collins-Thompson, K. (2017a). Optimizing search results for human learning goals. *Information Retrieval Journal*, *20*(5), 506–523.

Syed, R., & Collins-Thompson, K. (2017b). Retrieval algorithms optimized for human learning. In *Proceedings of the 40th international acm sigir conference on research and development in information retrieval* (pp. 555–564).

Syed, R., & Collins-Thompson, K. (2018). Exploring document retrieval features associated with improved short-and long-term vocabulary learning outcomes. In *Proceedings of the 2018 conference on human information interaction & retrieval* (pp. 191–200).

Tamine, L., & Chouquet, C. (2017). On the impact of domain expertise on query formulation, relevance assessment and retrieval performance in clinical settings. *Information Processing & Management*, *53*(2), 332–350.

Tamine, L., & Daoud, M. (2018). Evaluation in contextual information retrieval : Foundations and recent advances within the challenges of context dynamicity and data privacy. *ACM Computing Surveys (CSUR)*, *51*(4), 1–36.

Tamine-Lechani, L., Boughanem, M., & Daoud, M. (2010). Evaluation of contextual information retrieval effectiveness : overview of issues and research. *Knowledge and Information Systems*, *24*, 1–34.

Tamkin, A., Brundage, M., Clark, J., & Ganguli, D. (2021). Understanding the capabilities, limitations, and societal impact of large language models. *CoRR*, *abs/2102.02503*.

Tanudjaja, F., & Mui, L. (2002). Persona : A contextualized and personalized web search. In *Proceedings of the 35th annual hawaii international conference on system sciences* (pp. 1232–1240).

Teevan, J., Dumais, S. T., & Horvitz, E. (2005). Personalizing search via automated analysis of interests and activities. In *Proceedings of the 28th annual international acm sigir conference on research and development in information retrieval* (pp. 449–456).

Tetali, R., Bose, J., & Arif, T. (2013). Browser with clustering of web documents. In *2013 2nd international conference on advanced computing, networking and security* (pp. 164–168).

Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., & Gurevych, I. (2021). BEIR : A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth conference on neural information processing systems datasets and benchmarks track (round 2).*

Toker, D., Lallé, S., & Conati, C. (2017). Pupillometry and head distance to the screen to predict skill acquisition during information visualization tasks. In *Proceedings of the 22nd international conference on intelligent user interfaces* (pp. 221–231).

Urgo, K., & Arguello, J. (2022). Learning assessments in search-as-learning : A survey of prior work and opportunities for future research. *Information Processing & Management*, *59*(2), 102821.

Urgo, K., & Arguello, J. (2023). Goal-setting in support of learning during search : An exploration of learning outcomes and searcher perceptions. *Information Processing & Management*, *60*(2), 103158.

Vakkari, P. (2016). Searching as learning : A systematization based on literature. *J. Inf. Sci.*, *42*(1), 7–18.

Vakkari, P., Pennanen, M., & Serola, S. (2003). Changes of search terms and tactics while writing a research proposal : A longitudinal case study. *Information processing & management*, *39*(3), 445–463.

Van Leekwijck, W., & Kerre, E. E. (1999). Defuzzification : criteria and classification. *Fuzzy sets and systems*, *108*(2), 159–178.

van Lieshout, L. L., Vandenbroucke, A. R., Müller, N. C., Cools, R., & de Lange, F. P. (2018). Induction and relief of curiosity elicit parietal and frontal activity. *Journal of Neuroscience*, *38*(10), 2579–2588.

Van Meteren, R., & Van Someren, M. (2000). Using content-based filtering for recommendation. In *Proceedings of the machine learning in the new information age : Mlnet/ecml2000 workshop* (Vol. 30, pp. 47–56).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. In *Nips* (pp. 5998–6008).

Verma, M., Yilmaz, E., & Craswell, N. (2016). On obtaining effort based judgements for information retrieval. In *Proceedings of the ninth acm international conference on web search and data mining* (pp. 277–286).

von Hoyer, J., Pardi, G., Kammerer, Y., & Holtz, P. (2019). Metacognitive judgments in searching as learning (sal) tasks : Insights on (mis-) calibration, multimedia usage, and confidence. In *Proceedings of the 1st international workshop on search as learning with multimedia information* (pp. 3–10).

Voorhees, E. M. (2002). The philosophy of information retrieval evaluation. In *Evaluation of cross-language information retrieval systems : Second workshop of the cross-language evaluation forum, clef 2001 darmstadt, germany, september 3–4, 2001 revised papers 2* (pp. 355–370).

Vu, T., Nguyen, D. Q., Johnson, M., Song, D., & Willis, A. (2017). Search personalization with embeddings. In *European conference on information retrieval* (pp. 598–604).

Vu, T., Willis, A., Tran, S. N., & Song, D. (2015). Temporal latent topic user profiles for search personalisation. In *Advances in information retrieval : 37th european conference on ir research, ecir 2015, vienna, austria, march 29-april 2, 2015. proceedings 37* (pp. 605–616).

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). Minilm : Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in neural information processing systems* (Vol. 33, pp. 5776–5788).

Weller, M. (2011). *The digital scholar : How technology is transforming scholarly practice*. Bloomsbury Academic.

Wesche, M. B., & Paribakht, T. S. (1996). Assessing second language vocabulary knowledge : Depth versus breadth. *Canadian Modern Language Review-revue Canadienne Des Langues Vivantes*, *53*, 13–40.

White, R. W., Dumais, S. T., & Teevan, J. (2009). Characterizing the influence of domain expertise on web search behavior. In *Proceedings of the second acm international conference on web search and data mining* (pp. 132–141).

White, R. W., Ruthven, I., Jose, J. M., & Rijsbergen, C. V. (2005). Evaluating implicit feedback models using searcher simulations. *ACM Transactions on Information Systems (TOIS)*, *23*(3), 325–361.

Wildemuth, B. M. (2004). The effects of domain knowledge on search tactic formulation. *Journal of the american society for information science and technology*, *55*(3), 246–258.

Williams, K., & Zitouni, I. (2017). Does that mean you're happy ? rnn-based modeling of user interaction sequences to detect good abandonment. In *Proceedings of the 2017 acm on conference on information and knowledge management* (pp. 727–736).

Williams, M.-A. (1995a). Iterated theory base change : A computational model. In *Ijcai* (Vol. 95, pp. 1541–1547).

Williams, M.-A. (1995b, 01). Iterated theory base change : A computational model. In (p. 1541-1549).

Wilson, M. J., & Wilson, M. L. (2013). A comparison of techniques for measuring sensemaking and learning within participant-generated summaries. *Journal of the American Society for Information Science and Technology*, *64*(2), 291–306.

Wu, W.-C., Kelly, D., Edwards, A., & Arguello, J. (2012). Grannies, tanning beds, tattoos and nascar : Evaluation of search tasks with varying levels of cognitive complexity. In *Proceedings of the 4th information interaction in context symposium* (pp. 254–257).

Xie, I., & Joo, S. (2012). Factors affecting the selection of search tactics : Tasks, knowledge, process, and systems. *Information Processing & Management*, *48*(2), 254–270.

Xu, L., Zhou, X., & Gadiraju, U. (2020). How does team composition affect knowledge gain of users in collaborative web search ? In *Proceedings of the 31st acm conference on hypertext and social media* (pp. 91–100).

Yen, A.-Z., Huang, H.-H., & Chen, H.-H. (2019). Personal knowledge base construction from text-based lifelogs. In *Proceedings of the 42nd international acm sigir conference on research and development in information retrieval* (pp. 185–194).

Yilmaz, E., Verma, M., Craswell, N., Radlinski, F., & Bailey, P. (2014). Relevance and effort : An analysis of document utility. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management, CIKM 2014, shanghai, china, november 3-7, 2014* (pp. 91–100).

Yilmaz, Z. A., Wang, S., Yang, W., Zhang, H., & Lin, J. (2019). Applying BERT to document retrieval with birch. In *Emnlp/ijcnlp (3)* (pp. 19–24). Association for Computational Linguistics.

You, G.-w., & Hwang, S.-w. (2007). Personalized ranking : A contextual ranking approach. In *Proceedings of the 2007 acm symposium on applied computing* (pp. 506–510).

Yu, C., Liu, K.-L., Meng, W., Wu, Z., & Rishe, N. (2002). A methodology to retrieve text documents from multiple databases. *IEEE Transactions on Knowledge and Data Engineering*, *14*(6), 1347–1361.

Yu, R., Dietze, S., et al. (2022). Still haven't found what you're looking for–detecting the intent of web search missions from user interaction features. *arXiv preprint arXiv :2207.01256*.

Yu, R., Gadiraju, U., & Dietze, S. (2018). Detecting, understanding and supporting everyday learning in web search. *arXiv preprint arXiv :1806.11046*.

Yu, R., Gadiraju, U., Holtz, P., Rokicki, M., Kemkes, P., & Dietze, S. (2018a). Predicting user knowledge gain in informational search sessions. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*.

Yu, R., Gadiraju, U., Holtz, P., Rokicki, M., Kemkes, P., & Dietze, S. (2018b). Predicting user knowledge gain in informational search sessions. In *The 41st international acm sigir conference on research & development in information retrieval* (pp. 75–84).

Yu, R., Tang, R., Rokicki, M., Gadiraju, U., & Dietze, S. (2021a). Topic-independent modeling of user knowledge in informational search sessions. *Information Retrieval Journal*, *24*(3), 240–268.

Yu, R., Tang, R., Rokicki, M., Gadiraju, U., & Dietze, S. (2021b). Topic-independent modeling of user knowledge in informational search sessions. *Information Retrieval Journal*, *24*(3), 240–268.

Zamani, H., & Shakery, A. (2018). A language model-based framework for multi-publisher content-based recommender systems. *Information Retrieval Journal*, *21*, 369–409.

Zamir, O., & Grouper, O. E. (s. d.). A dynamic clustering interface to web search results. In *Proceedings of www8* (pp. 11–16).

Zeng, H.-J., He, Q.-C., Chen, Z., Ma, W.-Y., & Ma, J. (2004). Learning to cluster web search results. In *Proceedings of the 27th annual international acm sigir conference on research and development in information retrieval* (pp. 210–217).

Zhai, C., & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, *22*(2), 179–214.

Zhang, X., Anghelescu, H. G., & Yuan, X. (2005). Domain knowledge, search behaviour, and search effectiveness of engineering and science students : An exploratory study. *Information Research : An International Electronic Journal*, *10*(2), n2.

Zhang, X., Cole, M., & Belkin, N. (2011). Predicting users' domain knowledge from search behaviors. In *Proceedings of the 34th international acm sigir conference on research and development in information retrieval* (pp. 1225–1226).

Zhang, X., Liu, J., Cole, M., & Belkin, N. (2015). Predicting users' domain knowledge in information retrieval using multiple regression analysis of search behaviors. *Journal of the Association for Information Science and Technology*, *66*(5), 980–1000.

Zhang, Y., & Liu, C. (2020). Users' knowledge use and change during information searching process : A perspective of vocabulary usage. In *Proceedings of the acm/ieee joint conference on digital libraries in 2020* (pp. 47–56).

Zhou, Y., Dou, Z., & Wen, J.-R. (2020). Encoding history with context-aware representation learning for personalized search. In *Proceedings of the 43rd international acm sigir conference on research and development in information retrieval* (pp. 1111–1120).

Zumbach, J., & Mohraz, M. (2008). Cognitive load in hypermedia reading comprehension : Influence of text type and linearity. *Computers in human behavior*, *24*(3), 875–887.

List of Figures

List of Tables

List of Definitions

List of Examples

List of Algorithms

Représenter, Suivre et Evaluer les Connaissances et Besoins des Utilisateurs et leur Evolution dans le Cadre de la Recherche d'Information

Dima EL ZEIN

Résumé

L'utilisation de systèmes de recherche d'information est désormais un élément essentiel de notre quotidien, offrant une source d'information riche et facile d'accès. Ces systèmes, notamment les moteurs de recherche, peuvent maintenant fournir rapidement des données factuelles en adaptant les résultats en fonction de certains facteurs contextuels tels que la localisation, le type d'appareil et les intérêts de l'utilisateur. Cependant, ils ne sont optimisés, ni pour répondre aux objectifs d'apprentissage des utilisateurs, ni pour tenir compte de l'évolution de leurs connaissances. En effet, les connaissances de l'utilisateur ne sont pas statiques, mais bien dynamiques : l'utilisateur entame une session de recherche avec ses connaissances préexistantes et il continue d'en acquérir de nouvelles tout au long du processus de recherche. Lors de l'utilisation des outils de recherche actuels, l'utilisateur peut soumettre plusieurs requêtes et ainsi examiner de nombreux documents, dont certains peuvent manquer de pertinence et n'apporter aucune plus-value à sa connaissance. Cela peut entraîner une perte de temps et de motivation dans le processus de recherche d'informations souhaitées. C'est pour cela que le domaine de l'adaptation des systèmes de recherche pour l'apprentissage de connaissances, communément appelé "recherche comme apprentissage" ou "search as learning", a récemment fait l'objet d'une attention considérable. Cependant, nous pensons qu'avant de nous lancer dans l'adaptation de tels systèmes, il est tout d'abord indispensable de comprendre comment les utilisateurs apprennent et acquièrent des connaissances et de savoir comment les informations doivent être structurées dans les systèmes. Par ailleurs, l'évaluation de ces systèmes présente un réel défi car les méthodes de recherche et les mesures d'évaluation existantes négligent souvent la représentation et le suivi des connaissances de l'utilisateur. De plus, aucun jeu de données n'est disponible pour mesurer l'efficacité de ces systèmes qui prétendent aider à l'apprentissage pendant les sessions de recherche. Dans cette thèse, notre objectif est de surmonter ces problèmes en proposant différentes approches pour représenter les connaissances de l'utilisateur ainsi que ses objectifs d'apprentissage dans les systèmes de recherche. Nous proposons un cadre capable de suivre de manière dynamique l'évolution de ses connaissances et de ses besoins et d'estimer l'évolution de l'apprentissage tout au long de la session de recherche. Nous proposons ensuite une nouvelle mesure qui évalue dynamiquement les documents et les classent selon les besoins changeants de l'utilisateur et l'évolution de ses connaissances. Nous construisons également un jeu de données permettant de suivre l'évolution des connaissances de l'utilisateur tout au long des sessions de recherche. Ce jeu de données pourra servir de référence pour de futurs travaux. Enfin, nous proposons un cadre théorique pour implémenter ces concepts dans un système multi-agents doté de capacités de raisonnement basées sur des règles.

Mots-clés : Recherche d'information, Recherche comme Apprentissage, État de Connaissance, Objectifs d'Apprentissage, Évaluation, Agents BDI.

Abstract

The use of search tools has become an integral part of our daily lives, providing a readily accessible source of information and facilitating our acquisition of knowledge. However, while these tools are efficient in delivering factual data and adapting results based on contextual factors such as location, device, and interests, they are not optimized to support users' learning goals or account for their changing knowledge states. The user's knowledge is not static but