The Information Retrieval Series

Jianfeng Gao
Chenyan Xiong
Paul Bennett
Nick Craswell

Neural Approaches to Conversational Information Retrieval

Springer

The Information Retrieval Series

Volume 44

Information Retrieval (IR) deals with access to and search in mostly unstructured information, in text, audio, and/or video, either from one large file or spread over separate and diverse sources, in static storage devices as well as on streaming data. It is part of both computer and information science, and uses techniques from e.g. mathematics, statistics, machine learning, database management, or computational linguistics. Information Retrieval is often at the core of networked applications, web-based data management, or large-scale data analysis.

The Information Retrieval Series presents monographs, edited collections, and advanced text books on topics of interest for researchers in academia and industry alike. Its focus is on the timely publication of state-of-the-art results at the forefront of research and on theoretical foundations necessary to develop a deeper understanding of methods and approaches.

This series is abstracted/indexed in EI Compendex and Scopus.

Jianfeng Gao • Chenyan Xiong • Paul Bennett •
Nick Craswell

Neural Approaches to Conversational Information Retrieval

Springer

Jianfeng Gao
Microsoft
Redmond, WA, USA

Chenyan Xiong
Microsoft
Redmond, WA, USA

Paul Bennett
Microsoft
Redmond, WA, USA

Nick Craswell
Microsoft
Redmond, WA, USA

ISSN 1871-7500 ISSN 2730-6836 (electronic)
The Information Retrieval Series
ISBN 978-3-031-23079-0 ISBN 978-3-031-23080-6 (eBook)
https://doi.org/10.1007/978-3-031-23080-6

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

A conversational information retrieval (CIR) system is an information retrieval (IR) system with a conversational interface, which allows users to interact with the system to seek information via multi-turn conversations of natural language, in spoken or written form. Recent progress in deep learning has brought tremendous improvements in natural language processing (NLP) and conversational AI, leading to a plethora of commercial conversational services that allow naturally spoken and typed interaction, increasing the need for more human-centric interactions in IR. As a result, we have witnessed a resurgent interest in developing modern CIR systems in both research communities and industry.

This book surveys recent advances in CIR, focusing on neural approaches that have been developed in the last few years. In the year 2020, we gave a tutorial on CIR at SIGIR. It was well received and led to the invitation from Prof. ChengXiang Zhai to write a book on the topic for the information retrieval series of which he and Maarten de Rijke are the editors. After we exchanged a few emails, we reached a consensus that this is a timely exciting topic and the four of us can form an effective team for the project since we are passionate about CIR and had been working on it for years with complementary focuses and experiences. The primary target audience of the book are IR and NLP communities. However, audiences with other background, such as machine learning and human-computer interaction, will also find it an accessible introduction to CIR. We hope that this book will prove a valuable resource for students, researchers, and software developers.

Book Organization

The book contains nine chapters. Chapter 1 motivates the research of CIR by reviewing the studies on how people search, showing that information seeking can be cast in a framing of human-machine conversations. We then describe the properties of CIR, which lead to the definition of a CIR system and a reference architecture which we will describe in detail in the rest of the book. To provide the background for the discussions, we brief the early works in CIR.

In Chap. 2, we provide a detailed discussion of techniques for evaluating a CIR system—a goal-oriented conversational AI system with a human in the loop. We present two approaches. System-oriented evaluation captures the user's requests and preferences in a fixed dataset. User-oriented evaluation studies the interactions of a real user with the search system. Then, we describe two emerging forms of evaluation: CIR user simulation and responsible CIR.

Chapters 3 to 7 describe the algorithms and methods for developing main CIR modules (or sub-systems). In Chap. 3 we discuss conversational document search, which can be viewed as a sub-system of the CIR system. We start with an introduction to the task and public benchmarks; review transformation-based pre-trained language models, which are the building blocks of many CIR modules; and then describe the main components of the conversational search system, including contextual query understanding, sparse and dense document retrieval, and neural document ranking.

In Chap. 4, we discuss algorithms and methods for query-focused multi-document summarization, which aim at producing a concise summary of a set of documents returned by the document search module in response to a user query. This is one of the key components of the result generation module of a CIR system.

In Chap. 5, we describe various neural models for conversational machine comprehension (CMC), which generate a direct answer to a user query based on retrieved query-relevant documents. Equipped with CMC, a CIR system can be used as an open-domain conversational question answering system.

In Chap. 6, we discuss neural approaches to conversational question answering over knowledge bases (C-KBQA), which is fundamental to the knowledge base search module of a CIR system. We introduce the C-KBQA task, describe the forms of knowledge bases and the open benchmarks, and discuss in detail a modular C-KBQA system that is based on semantic parsing. We then present a unitary (non-modular) system that is based on a transformer-based language model, which unifies the C-KBQA modules.

In Chap. 7, we discuss various techniques and models that aim to equip a CIR system with the capability of proactively leading a human-machine conversation by asking a user to clarify her search intent, suggesting the user what to query next, or recommending a new topic for the user to explore. We end this chapter with a brief survey of conversational recommendation systems.

In Chap. 8, we review a variety of commercial systems for CIR and related tasks. We first present an overview of research platforms and toolkits, which enable scientists and practitioners to build conversational experiences. Then we review historical highlights and recent trends in a range of application areas.

Chapter 9 concludes the book with a brief discussion of research trends and areas for future work.

Acknowledgments

Many people have supported us and provided valuable feedback to the writing of this book, namely, but not limited to W. Bruce Croft, Hao Cheng, Bill Dolan, Susan Dumais, Michel Galley, Jimmy Lin, Jian-Yun Nie, Xiaodong Liu, Zhiyuan Liu, Baolin Peng, Dragomir Radev, Corby Rosset, Ryen White, Huiyuan Xie, Shi Yu, and Zhou Yu.

We would also like to thank the anonymous reviewer for the insightful feedback and Ralf Gerstner and ChengXiang Zhai for making the publication of this book possible.

Redmond, WA, USA

Jianfeng Gao
Chenyan Xiong
Paul Bennett
Nick Craswell

Contents

Chapter 1
Introduction

A conversational information retrieval (CIR) system is an information retrieval (IR) system with a conversational interface, which allows users to interact with the system to seek information via multi-turn conversations of natural language (in spoken or written form). CIR provides a more natural user interface for information seeking than traditional, single-turn, search engines and is particularly useful for search on modern devices with small or no screen.

CIR is a long-standing topic, which we can trace back to the 1960s. However, the research in CIR remained in its infancy until the 2010s due to the lack of large amounts of conversational data, sufficient natural language processing (NLP) technologies, strong commercial needs, etc. Even today, popular commercial search engines, such as Google and Bing, provide only limited dialog capabilities.

Recent progress in machine learning (e.g., deep learning) has brought tremendous improvements in NLP and conversational AI, leading to a plethora of commercial conversational services that allow naturally spoken and typed interaction, increasing the need for more human-centric interactions in IR. As a result, we have witnessed a resurgent interest in developing modern CIR systems in both research communities and industry.

This book surveys recent advances in CIR, focusing mainly on neural approaches that have been developed in the last 10 years. The primary target audience of the book are IR and NLP communities. However, audiences with other background, such as machine learning and human-computer interaction, will also find it an accessible introduction to CIR. We hope that this book will prove a valuable resource for students, researchers, and software developers.

In the rest of this chapter, we first review recent survey paper on CIR and motivate the research of CIR by reviewing the studies on how people search, showing that information seeking can be cast in a framing of human-machine conversations. We then describe the properties of CIR, which lead to the definition of a CIR system (Radlinski and Craswell 2017) and a reference architecture (Fig. 1.4), which we will describe in detail in the rest of the book. To provide the background for the discussions, we brief the early works in CIR (Croft 2019).

J. Gao et al., *Neural Approaches to Conversational Information Retrieval*,
The Information Retrieval Series 44, https://doi.org/10.1007/978-3-031-23080-6_1

1.1 Related Surveys

CIR, and conversational AI in general, is a multidisciplinary and broad research area, attracting researchers from multiple communities, including IR, NLP, speech, dialog, human computer interaction, deep learning, etc. Although many surveys and tutorials on dialog, question answering (QA), and neural approaches to conversational AI have been published in recent years, to the best of our knowledge, this is the first book that dedicates to neural approaches to CIR.

The SWIRL 2018 workshop report (Culpepper et al. 2018) presents a summary of open changes in IR and considers CIR as one of the most important emerging research areas in IR.

Gao et al. (2019) present an overview of the state-of-the-art deep learning approaches developed for three types of dialog systems: question answering, task-oriented dialogue, and social chat bots. CIR is related to but differs significantly from the three types of dialog systems. For example, in CIR, a user often starts with a search goal (similar to that of QA and task-oriented dialog systems) but then shifts her interest and starts to explore new topics based on the result returned by CIR (similar to that of social chatbots). This books presents a comprehensive view of CIR by consolidating task definitions and problem formulations of previous works.

Zamani et al. (2022) provide a different and complementary overview of CIR, focusing on defining the CIR research area and drawing connection between its subareas (e.g., conversational search, QA, and recommendation). This book focuses more on the development and evaluation of deep learning models for CIR problems.

Wu and Yan (2019) review deep learning models that have been developed for chitchat dialog systems. Zhou et al. (2020) present an in-depth discussion of the techniques behind Microsoft's chatbot, Xiaoice. Su et al. (2018a) review goal-oriented spoken dialogue systems, focusing on dialog component techniques such as spoken utterance understanding, state tracking, and dialogue management. Wen et al. (2019) survey the datasets developed for conversational AI.

One of the impetuses for writing this book came in the summer of 2020, when the authors gave a tutorial on CIR at SIGIR (Gao et al. 2020b). The tutorial was well received and led to the invitation from Prof. ChengXiang Zhai to write a book on the topic for the information retrieval series.

1.2 How People Search

This section reviews search tasks and theoretical models of IR. A good survey of early works is presented by Marti Hearst in Chapter 2 of Baeza-Yates et al. (2011). A discussion on recent works is reported in Collins-Thompson et al. (2017).

1.2.1 Information-Seeking Tasks

People search for various reasons, ranging from looking up disputed news or a weather report to completing complex tasks such as hotel booking or travel planning.

Marchionini (2006) group information-seeking tasks into two categories, *information lookup* and *exploratory search*. Lookup tasks are akin to factoid retrieval or question answering tasks, which modern Web search engines and standard database management systems are largely engineered and optimized to fulfill.

Exploratory tasks include complex search tasks, such as learning and investigating searches. Compared to lookup tasks, exploratory searches require a more intensive human-machine interaction over a longer-term iterative *sensemaking* process (Pirolli and Card 2005) of formulating a conceptual representation from a large collection of information. For example, learning searches involve users reading multiple information items and synthesizing content to form new understanding. In investigating searches, such as travel planning and academic research, users often take multiple iterations over long periods of time to collect and access search results before forming their personal perspectives of the topics of interest.

Despite that more than a quarter of Web search are complex (Collins-Thompson et al. 2017), modern Web search engines are not optimized for complex search tasks. Since in these tasks human users heavily interact with information content, the search engine needs to be designed as an intelligent task-oriented conversational system of facilitating the communication between users and content to achieve various search tasks. CIR systems we discussed in this book are mainly developed for complex searches.

1.2.2 Information-Seeking Models

Many theoretical models of how people search have been developed to help improve the design of IR systems. A classic example is the cognitive model of IR proposed by Sutcliffe and Ennis (1998), where the information-seeking process is formulated as a cycle consisting of four main activities:

1. problem identification,
2. articulation of information needs,
3. query (re-)formulation, and
4. results evaluation.

Early models mainly focus on information lookup tasks, which often do not require multiple query-response turns. These models assume that the user's information need is static, and the information-seeking process is one of successively refining a query until enough relevant documents have been retrieved.

More recent models emphasize the dynamic nature of the search process, as observed in exploratory searches, where users learn as they search and adjust their information needs as they read and evaluate search results. A typical example is the *berry-picking* model (Bates 1989), which presents a dynamic search process, where a berry-picker (searcher) may issue a quick, under-specified query in the hope of getting into approximately the right part of the information space or simply to *test the water* and then reformulate her query to be more specific to get closer to the information of interest.

Some information-seeking models focus on modeling the search strategy or policy that controls the search process. For example, Bates (1979) suggests that searchers' behaviors can be formulated as a hierarchical decision-making process, which is characterized by search strategies (high-level policies), which in turn are made up of sequences of search tactics (low-level policies) and that searchers often monitor the search process, evaluate the cost and benefit of each step, and adjust the policies to optimize the return. Bates's model bears a strong resemblance to the cognitive model that motivates the development of the classic modular architecture of task-oriented dialog systems illustrated in Fig. 1.1, which will be discussed next.

The most relevant to CIR discussed in this book is the theoretical framework for conversational search, proposed by Radlinski and Craswell (2017). Summarizing all the requirements of CIR, they propose five properties to measure the extent to which an IR system is conversational. These properties are:

1. User revealment: The system helps the user express or discover her information need and long-term preferences.
2. System revealment: The system reveals to the user its capabilities and corpus, building the user's expectations of what it can and cannot do.
3. Mixed initiative: The system and user both can take initiative as appropriate.
4. Memory: The user can reference past statements, which implicitly also remain true unless contradicted.
5. Set retrieval: The system can reason about the utility of sets of complementary items.

Then, taking together these properties, they define a CIR system as a task-oriented dialog system "for retrieving information that permits a mixed-initiative back and forth between a user and agent, where the agent's actions are chosen in response to a model of current user needs within the current conversation, using both short- and long-term knowledge of the user."

1.3 CIR as Task-Oriented Dialog

A CIR process can be viewed as a task-oriented dialog, with information seeking as its task. This section describes the mathematical model and the classical modular architecture of task-oriented dialog systems, reviews how popular search engines support human-system interactions in Web search through the lens of a task-oriented

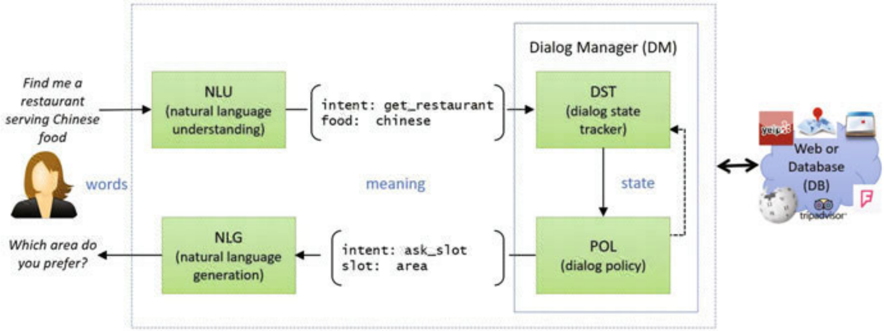
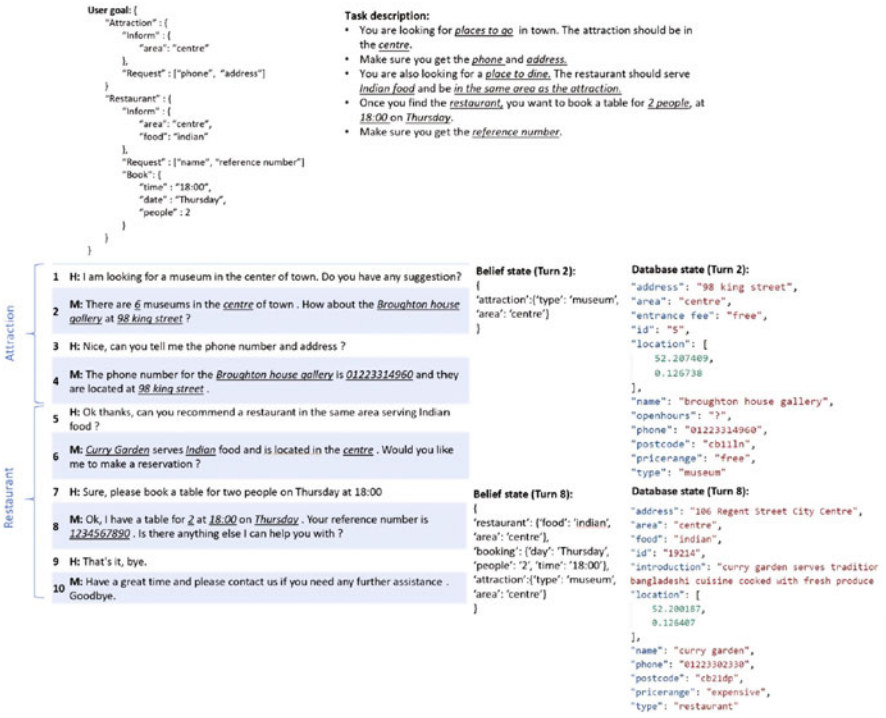

Find me a restaurant serving Chinese food
words
Which area do you prefer?
NLU (natural language understanding)
intent: get_restaurant
food: chinese
meaning
NLG (natural language generation)
intent: ask_slot
slot: area
Dialog Manager (DM)
DST (dialog state tracker)
state
POL (dialog policy)
Web or Database (DB)
yelp
tripadvisor

Fig. 1.1 A modular architecture for multi-turn task-oriented dialog systems. It consists of the following modules: NLU (natural language understanding), DM (dialog manager), and NLG (natural language generation). DM contains two sub-modules, DST (dialog state tracker) and POL (dialog policy). The dialog system, indicated by the dashed rectangle, has access to an external database or Web collection. Adapted from Gao et al. (2019)

dialog system, and summarizes the research topics being actively studied to make IR systems more conversational.

The classical modular approach to building task-oriented dialog systems (or task bots) is motivated by the theories of human cognition. Cognition is formulated as an iterative decision-making process (Marcus 2020): organisms (e.g., humans) take in information from the environment; build internal cognitive models based on their perception of that information, which includes information about the entities in the external world, their properties, and relationships; and then make decisions with respect to these cognitive models, which lead to human actions that change the environment. Cognitive scientists generally agree that the degree to which an organism prospers in the world depends on how good those internal cognitive models are (Gallistel 1990; Gallistel and King 2011).

Similarly, the classical modular architecture of task bots, as shown in Fig. 1.1, views multi-turn conversations between a system and a human user as an iterative decision-making process, where the system is (the agent of) the organism and the user the environment. The system consists of a pipeline of modules that play different roles in decision-making. At each iteration, a natural language understanding (NLU) module identifies the user intent and extracts associated information such as entities and their values from user input. A dialog state tracker (DST) infers the dialog belief state (the internal cognitive model of the dialog system). The belief state is often used to query a task-specific database (DB) to obtain the DB state, such as the number of entities that match the user goal. The dialog state and DB state are then passed to a dialog policy (POL) to select the next system action. A natural language generation (NLG) module converts the action to a natural language response. Like cognitive scientists, dialog researchers also believe that the quality of a task bot depends to a large degree upon the performance of dialog state tracking (or its internal cognitive model), which had been the focus of task-oriented dialog research for many years (e.g., Gao et al. 2019; Young et al. 2013).



User goal:

```
User goal: {
    "Attraction" : {
        "Inform" : {
            "area": "centre"
        },
        "Request" : ["phone",  "address"]
    }
    "Restaurant" : {
        "Inform" : {
            "area": "centre",
            "food": "indian"
        },
        "Request" : ["name", "reference number"]
        "Book": {
            "time" : "18:00",
            "date" : "Thursday",
            "people" : 2
        }
    }
}
```

Task description:

- You are looking for *places to go* in town. The attraction should be in the *centre*.
- Make sure you get the *phone* and *address.*
- You are also looking for a *place to dine.* The restaurant should serve *Indian food* and be *in the same area as the attraction.*
- Once you find the *restaurant,* you want to book a table for *2 people*, at *18:00* on *Thursday*.
- Make sure you get the *reference number*.

1 **H:** I am looking for a museum in the center of town. Do you have any suggestion?

2 **M:** There are *6* museums in the *centre* of town . How about the *Broughton house gallery* at *98 king street* ?

3 **H:** Nice, can you tell me the phone number and address ?

4 **M:** The phone number for the *Broughton house gallery* is *01223314960* and they are located at *98 king street* .

5 **H:** Ok thanks, can you recommend a restaurant in the same area serving Indian food ?

6 **M:** *Curry Garden* serves *Indian* food and is located in the *centre* . Would you like me to make a reservation ?

7 **H:** Sure, please book a table for two people on Thursday at 18:00

8 **M:** Ok, I have a table for *2* at *18:00* on *Thursday* . Your reference number is *1234567890* . Is there anything else I can help you with ?

9 **H:** That's it, bye.

10 **M:** Have a great time and please contact us if you need any further assistance . Goodbye.

Belief state (Turn 2):

```
{
'attraction':{'type': 'museum',
'area': 'centre'}
}
```

Belief state (Turn 8):

```
{
'restaurant': {'food': 'indian',
'area': 'centre'},
'booking': {'day': 'Thursday',
'people': '2', 'time': '18:00'},
'attraction':{'type': 'museum',
'area': 'centre'}
}
```

Database state (Turn 2):

```
"address": "98 king street",
"area": "centre",
"entrance fee": "free",
"id": "5",
"location": [
    52.207409,
    0.126738
],
"name": "broughton house gallery",
"openhours": "?",
"phone": "01223314960",
"postcode": "cb11ln",
"pricerange": "free",
"type": "museum"
```

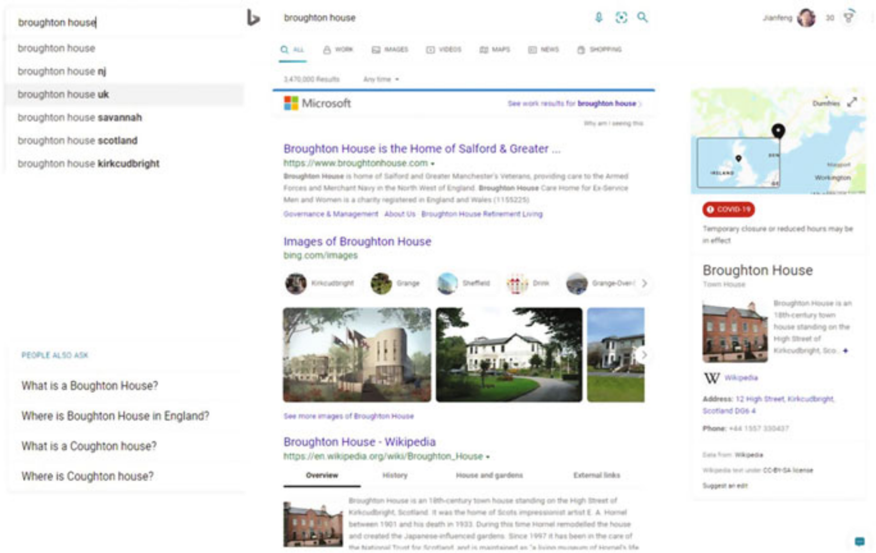
Database state (Turn 8):

```
"address": "106 Regent Street City Centre",
"area": "centre",
"food": "indian",
"id": "19214",
"introduction": "curry garden serves traditior
bangladeshi cuisine cooked with fresh produce
"location": [
    52.200187,
    0.126407
],
"name": "curry garden",
"phone": "01223302330",
"postcode": "cb21dp",
"pricerange": "expensive",
"type": "restaurant"
```

Fig. 1.2 An example of a task-oriented dialog. (Top) A user goal and a task description. (Bottom) Multiple turns of user-system utterances and the dialog belief states and database states at Turns 2 and 8. Adapted from Gao et al. (2020a)

Figure 1.2 is a dialog of completing a multi-domain task produced by a user and a dialog system (Gao et al. 2020a; Peng et al. 2020a). The user starts the conversation by asking for a recommendation of a museum in the center of town. The system identifies the user request and provides a recommendation based on the search result from an attraction DB. Then, the user wants to book a table in a restaurant in the same area. We can see that through the conversation, the system develops belief states, which can be viewed as the system's understanding of what the user needs and what is available in the DB. Based on belief state, the system picks the next action, either asking for clarification or providing the user with information being requested. This example also presents some challenges in conversational search. For example, the agent needs to understand that the "same area" refers to "center of town" (i.e., the so-called co-reference resolution problem) and then identifies a proper entity from the restaurant-booking DB to make the reservation.

The example shows that in a task-oriented dialog, the user and the system play different roles. The user knows (approximately) what she needs, but not what is available (in the DB). The system, on the other hand, knows what is available, but not the user's information needs. Dialog is a two-way process in which the user and system get to know each other to make a deal.



broughton house

- broughton house
- broughton house **nj**
- broughton house **uk**
- broughton house **savannah**
- broughton house **scotland**
- broughton house **kirkcudbright**

PEOPLE ALSO ASK

- What is a Boughton House?
- Where is Boughton House in England?
- What is a Coughton house?
- Where is Coughton house?

broughton house

3,470,000 Results Any time

Microsoft See work results for **broughton house**

Why am I seeing this

Broughton House is the Home of Salford & Greater ...
https://www.broughtonhouse.com
Broughton House is home of Salford and Greater Manchester's Veterans, providing care to the Armed Forces and Merchant Navy in the North West of England. **Broughton House** Care Home for Ex-Service Men and Women is a charity registered in England and Wales (1155225)
Governance & Management · About Us · Broughton House Retirement Living

Images of Broughton House
bing.com/images

See more images of Broughton House

Broughton House - Wikipedia
https://en.wikipedia.org/wiki/Broughton_House
Broughton House is an 18th-century town house standing on the High Street of Kirkcudbright, Scotland. It was the home of Scots impressionist artist E. A. Hornel between 1901 and his death in 1933. During this time Hornel remodelled the house and created the Japanese-influenced gardens. Since 1997 it has been in the care of the National Trust for Scotland, and is maintained as "a living museum of Hornel's life

COVID-19
Temporary closure or reduced hours may be in effect

Broughton House
Town House
Broughton House is an 18th-century town house standing on the High Street of Kirkcudbright, Sco...

Wikipedia

Address: 12 High Street, Kirkcudbright, Scotland DG6 4
Phone: +44 1557 330437

Data from: Wikipedia
Wikipedia text under CC-BY-SA license
Suggest an edit

Fig. 1.3 An example of Bing Web search interface. The system makes search more effective by autocompleting an input query (top left), organizing search results in the SERP (right), and suggesting related queries that people also ask (bottom left)

Now, consider the user-system interactions in Web search. The problem setting resembles that of task-oriented dialogs. The user knows (approximately) what she needs, but not what is available (on the Web). The system, on the other hand, knows what is available, but not the user's search intent. Unlike the dialog system demonstrated in Fig. 1.2, most popular search engines mainly treat Web search as a one-way information retrieval process. In the process, the user plays a proactive role to iteratively issue a query, inspect search results, and reformulate the query; while the system, taking the Bing Search engine as an example as illustrated in Fig. 1.3, plays a passive role to make search more effective by autocompleting a query, organizing search results in Search Engine Results Pages (SERP), and suggesting related queries that people also ask.

It is generally agreed that effective information seeking requires multi-turn user-system interactions where both parties can take initiative as appropriate and that users' search experiences can be significantly improved, especially on devices with small or no screen, if the system can play a more active role. Therefore, there have been many studies that explicitly model and support the interaction by tracking belief state (user intent), asking clarification questions, providing recommendations, understanding natural language input, generating natural language output, and so on. In this book, we will discuss methods and technologies, with a focus on neural approaches developed in the last ten years, which can be incorporated into IR systems to make search experiences more conversational, effortless, and enjoyable. We start our discussion with a reference architecture of CIR systems in the next section.

1.4 CIR System Architecture

The development of CIR systems is more challenging than building typical task bots because information seeking is an open-domain task while most task bots, whose architecture is shown in Fig. 1.1, are designed to perform domain-specific tasks.

The dialog in Fig. 1.2 is domain-specific, consisting of two domains, attraction-booking and restaurant-book. For each domain, a set of slots are defined by domain experts. In the restaurant-booking domain, for example, slots like `restaurant-name`, `location`, `food-type`, `number-people`, `phone-number`, `date`, `time`, etc. are necessary. Such a domain-specific dialog can be viewed as a process of *slot-filling*, where a user specifies the values for some slots to constrain the search, such as `location` and `food-type`, and the system tries to look for entities (restaurants) in its DB which meet the constraints and fills the slots whose values are asked by the user such as `restaurant-name`. Since the slots are pre-defined, the possible actions that the task bot can take at each dialog turn can also be pre-defined. For example, the system response in Turn 6 in the dialog of Fig. 1.2

“Curry Garden serves Indian food.”

is generated from the action template defined in the form of dialog act (Austin 1975) as:

$$\texttt{inform(restaurant} - \texttt{name} = \text{“}\texttt{...}'', \texttt{food} - \texttt{type} = \text{“}\texttt{...}'').$$

A CIR system, however, deals with open-domain dialogs with a much larger (or infinite) action space since users might search any information by issuing free-form queries. As a result, it is impossible for system designers to pre-define a set of actions that the system can take. Instead, we group actions into several action classes based on high-level user intents, such as asking clarifying questions, document search, shifting topics, and developing an action module for each class to generate responses.

Figure 1.4 shows a reference architecture of CIR systems that we will describe in detail in this book. The architecture is not only a significant extension of the task-oriented dialog system architecture in Fig. 1.1 to deal with open-domain information-seeking tasks but also an extension to popular Web search engines in that it explicitly models multi-turn user-system conversations (e.g., via dialog manager modules). It consists of three layers: the CIR engine layer, the user experience layer, and the data layer.

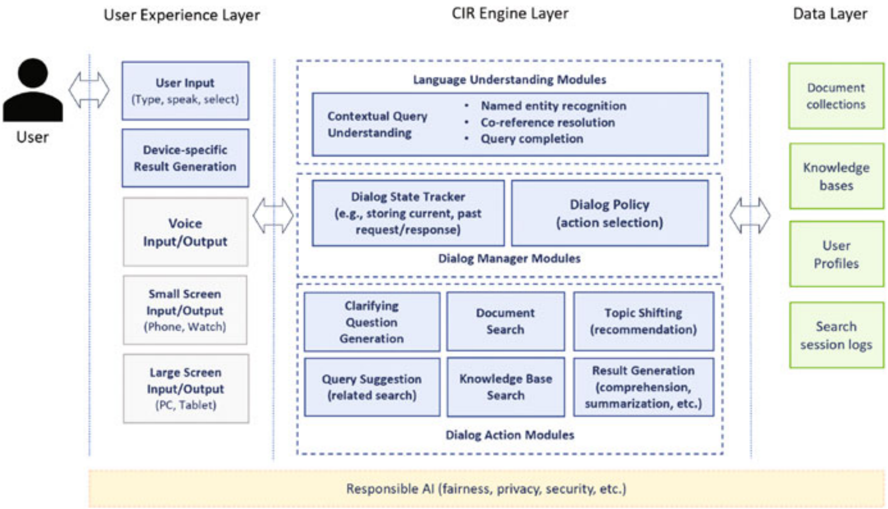

User Experience Layer
CIR Engine Layer
Data Layer
User
User Input (Type, speak, select)
Device-specific Result Generation
Voice Input/Output
Small Screen Input/Output (Phone, Watch)
Large Screen Input/Output (PC, Tablet)
Language Understanding Modules
Contextual Query Understanding
• Named entity recognition
• Co-reference resolution
• Query completion
Dialog State Tracker (e.g., storing current, past request/response)
Dialog Policy (action selection)
Dialog Manager Modules
Clarifying Question Generation
Document Search
Topic Shifting (recommendation)
Query Suggestion (related search)
Knowledge Base Search
Result Generation (comprehension, summarization, etc.)
Dialog Action Modules
Document collections
Knowledge bases
User Profiles
Search session logs
Responsible AI (fairness, privacy, security, etc.)

Fig. 1.4 A reference architecture of CIR systems

1.4.1 CIR Engine Layer

This layer consists of all the major modules of the system. We group them into three categories. The first category is a set of language understanding modules for contextual query understanding. Unlike the NLU module of task bots that performs slot-filling based on a pre-defined set of slots, contextual query understanding conceptually acts as a query rewriter. In conversational search, ellipsis phenomena are frequently encountered. The rewriter uses contextual information of dialog history (within the same search session) to rewrite the user input query at each dialog turn to a *de-contextualized* query, which can be used to retrieve relevant documents via calling search APIs[1] or retrieve answers from a knowledge base. These modules need to identify common types of name entities (e.g., person names, places, locations, etc.), replace pronouns with their corresponding entity mentions in a query (co-reference resolution), and complete the query. As shown in Fig. 1.5, user queries are rewritten to include context by, for example, replacing "him" in Turn 3 with the detected entity name "Ashin," "that" with "The Time Machine" in Turn 7, and adding "send The Time Machine" in Turn 9.

The second category is a set of dialog manager (DM) modules. Similar to DM in task bots, it consists of a dialog state tracker, which keeps track of the current dialog state (e.g., by storing and encoding search interactions, including current and past user requests and system responses, in working memory), and a dialog policy,

[1] The search API of most commercial search engines (e.g., Bing) only takes a single query, not a dialog session, as input.