

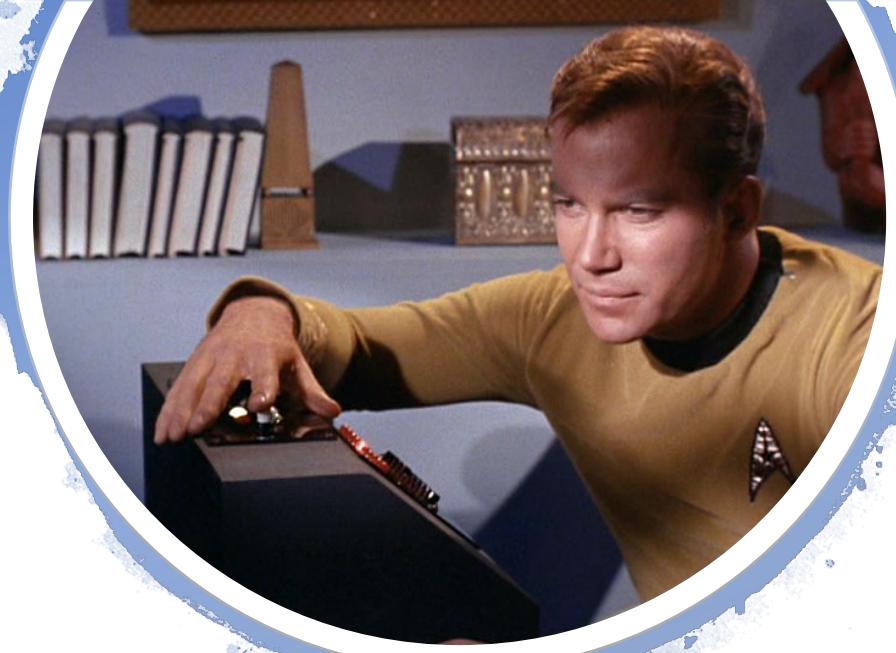
Conversational Information Seeking: Theory and Application

 #TheCISTutorial

Part 2: Historical Context



CIS has roots in early
(interactive) IR research!



Five general types of information necessary to get the desired state:

- Determination of subject
- Objective and motivation
- Personal characteristics of the inquirer
- Relationship of inquiry description to file organization
- Anticipated or acceptable answers

“Question Negotiation and Information
Seeking in Libraries”, Robert Taylor, 1968.

Intermediary-based Online IR

Nature of interaction between user and intermediary, in both cognitive and affective senses, is a key factor in search satisfaction.

User-Intermediary Interactions

I Alright. Right. The form ... err, what we got on the form just

U

I says community education in developing countries. That's

U

Yeah

I approximately yeah/1/ Can you tell me, sort of quite a lot

U yeah

well (illeg.) /2/

I more about what it is you're going to do-/3a/ it is your disser-

U

I tation/3b/ Yeah yeah. Tell me what . sorts of

U Yeah, M.A. dissertation/4/

I things you're going to do in your dissertation,/5a/ and then the

U

I sorts of things you want to rea::d./5b/

U

That's my sort of, uhm, plan,

I = Intermediary U = User /n/ = utterance number

5

Figure 3. Example span of user-intermediary pre-search interaction.

User-Intermediary Interactions

| VARIABLE | CODE |
|---|------|
| Problem Description function | PD |
| Mode: Ask | 1 |
| Offer | 2 |
| Task: Description of problem topic or clarification | a |
| Description of problem type or context | b |
| Description of subject area | c |
| Terminology | d |
| Literature | e |
| User Model function | UM |
| Mode: Ask | 1 |
| Offer | 2 |
| Task: Personal background (status) | a |
| Intentions/goals | b |
| Familiarity with information systems | c |
| Familiarity with topic | d |
| Knowledge resource codes | |
| Subject knowledge | SUB |
| System knowledge | SYS |
| Query language | 1 |
| Index language | 2 |
| Data base | 3 |
| Technical | 4 |
| Functions | 5 |
| Knowledge of users | US |
| Types | 1 |
| Characteristics | 2 |
| Intentions | 3 |
| General world knowledge | WK |

The THOMAS System

THOMAS, THE REFERENCE RETRIEVAL PROGRAM

Help can be obtained whenever the program has displayed the start symbol by typing ‘?’ immediately after it.

Please give a short name for the search:

► Alv.Resp.

Start searching:

► pulmonary alveoli

Influence of fasting on blood gas tension, pH, and related values in dogs.;
Pickrell *et al*, *Am J Vet Res*, 34, 805-8, Jun 73

1. J A Pickrell, 2. JL Mauderly, 3. B A Muggenburg, 4. U C Luft, 5. animal experiments, 6. animal feed, 7. arteries, 8. blood, 9. body temperature, 10. carbon dioxide, 11. dogs, 12. fasting, 13. hemoglobin, 14. hydrogen-ion concentration, 15. irrigation, 16. lung, 17. oxygen, 18. pulmonary alveoli, 19. respiration, 20. time factors

► ?

The THOMAS System

There can be three parts to your statement (all optional):

1. Your reaction to the reference just shown (if any).

This must come first:

"Yes" or "No"

2. A selection from the names (authors) or terms shown, by number. A "not" in the statement signifies rejection of all numbers that follow it.
3. New names or terms (terms preferably in quotes). The elements of the statement should be separated by commas.

Examples: 'posture', 'circulatory system'

Yes, not 11,12

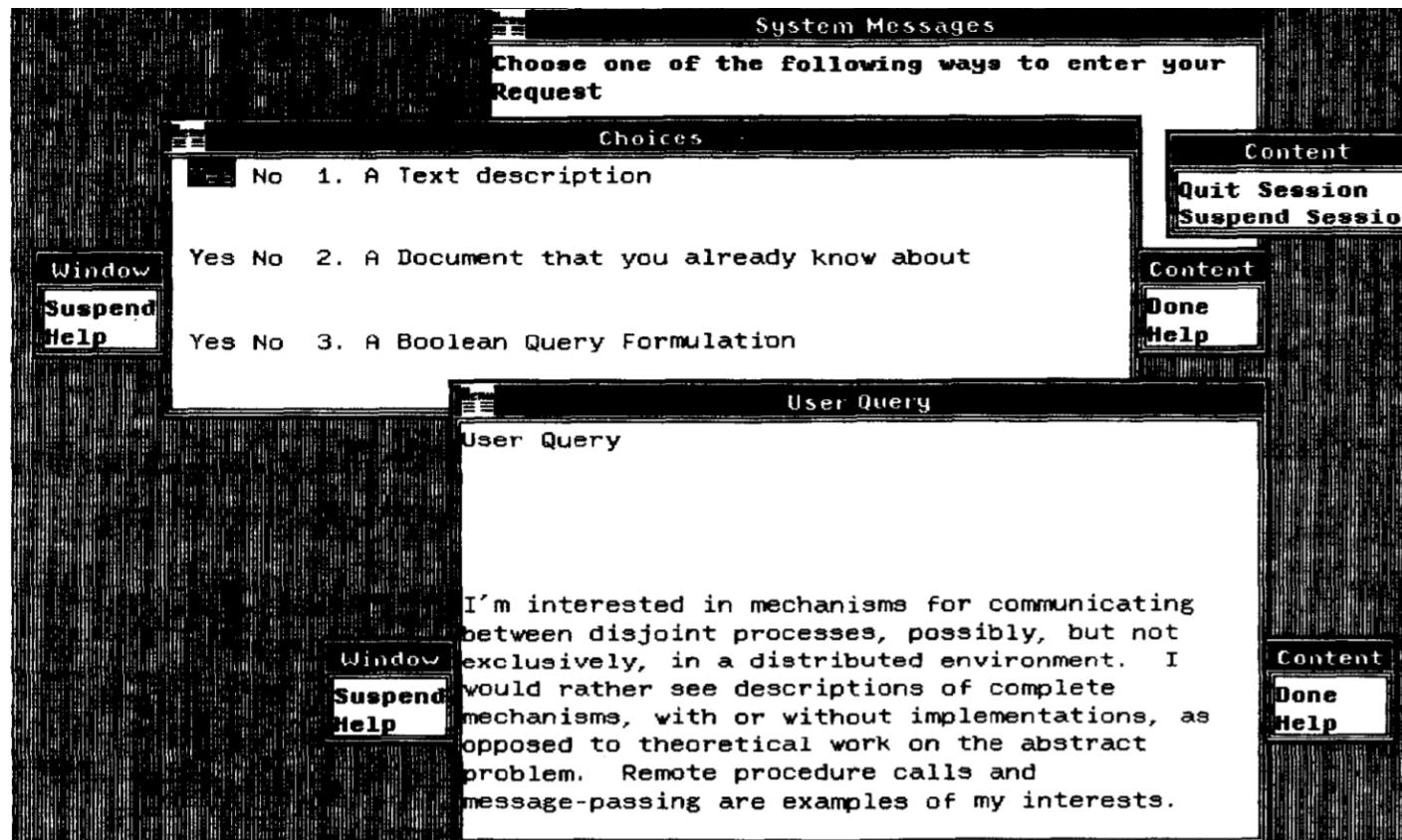
No, 7,13,4

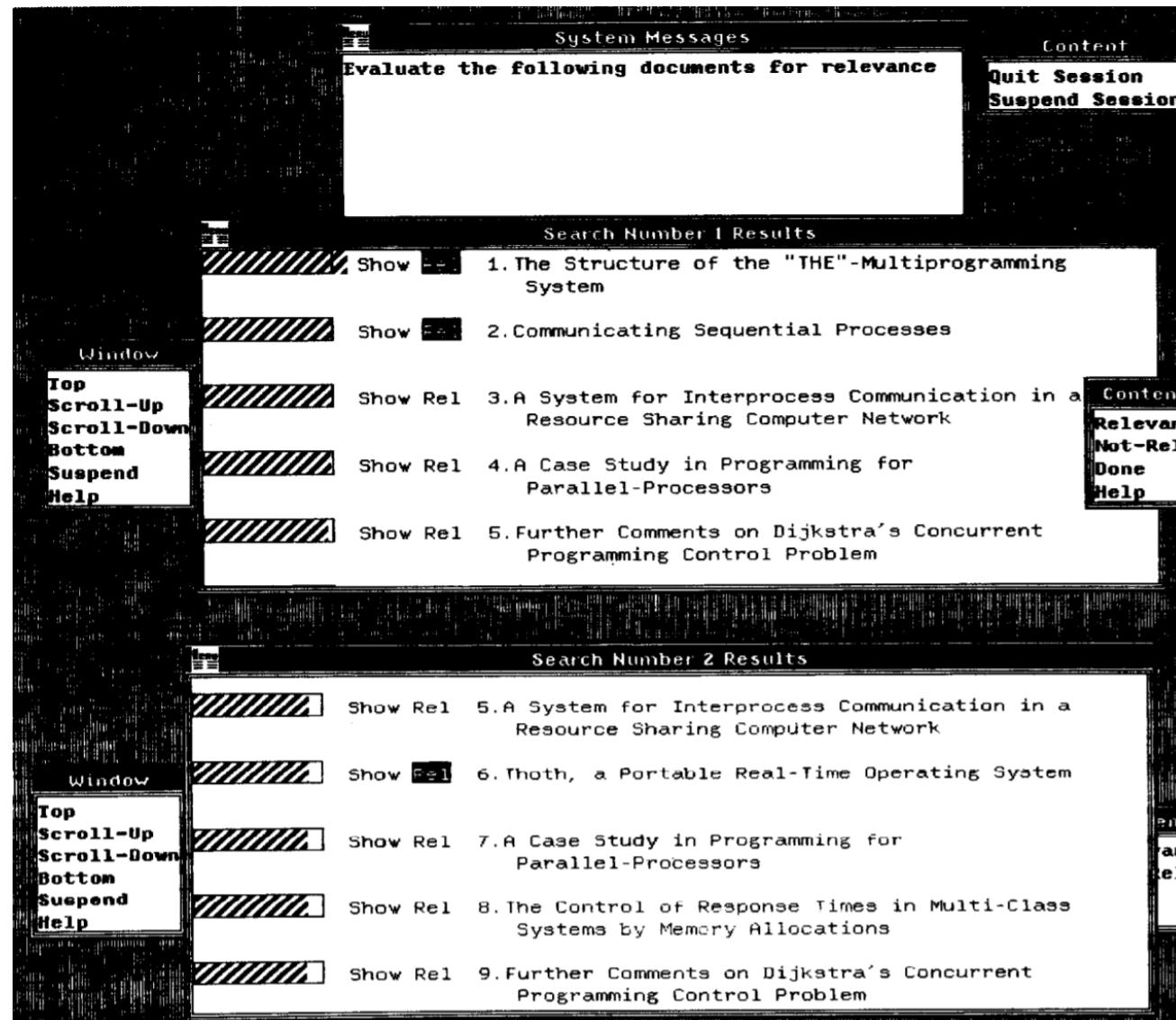
'heart rate'

Yes

Press enter key when you are ready to proceed ►

The I³R System





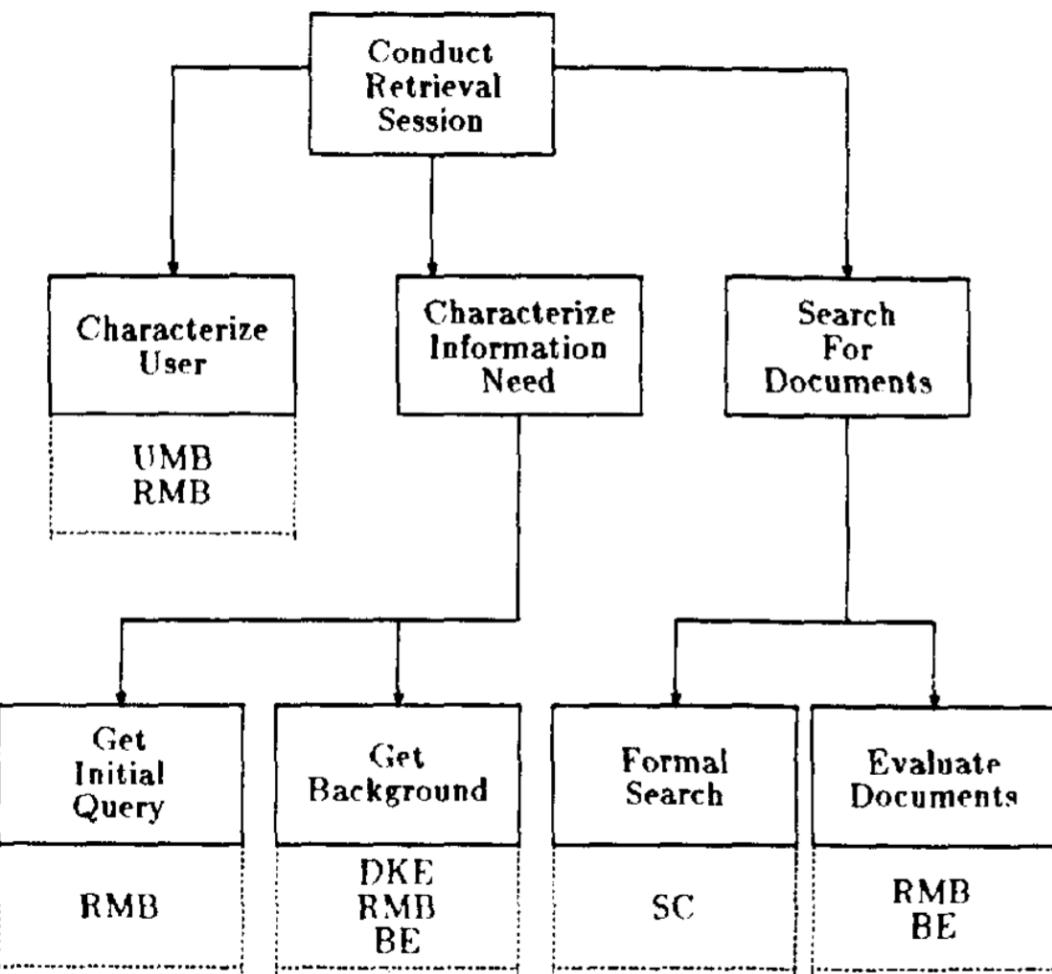


FIG. 3. Default scheduler plan.

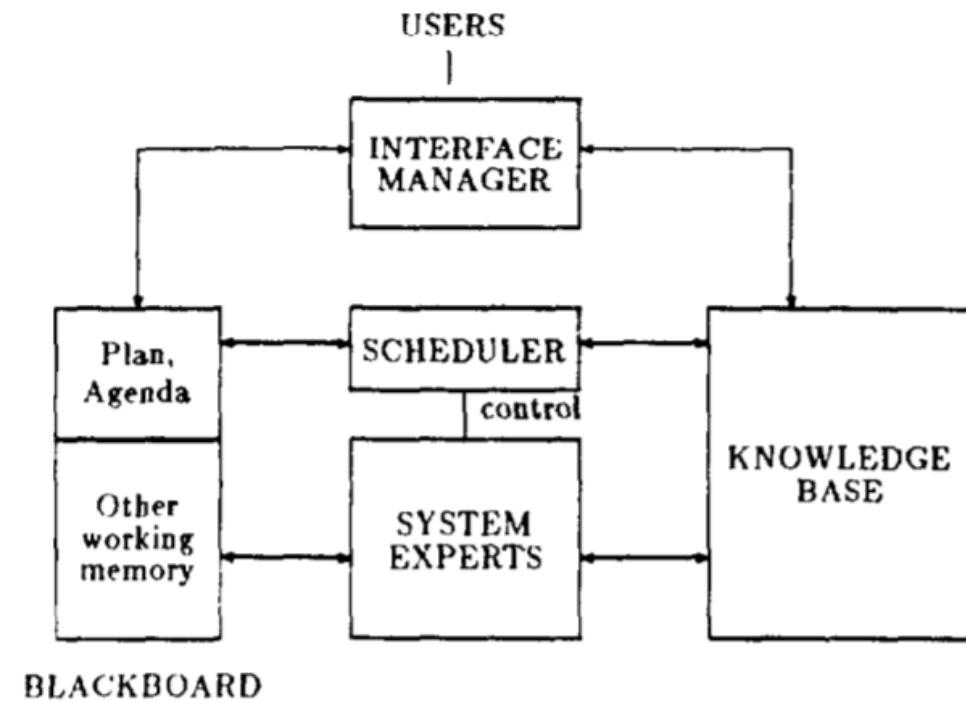


FIG. 2. Major system components.

Web Search Era!

Understanding User Behavior Through Log Data and Analysis

(Version of Apr 23, 2013)

Susan Dumais (Microsoft)
Robin Jeffries (Google)
Daniel M Russell (Google)
Diane Tang (Google)
Jaime Teevan (Microsoft)

WORKSHOP REPORT

SIGIR 2009 Workshop on Understanding the User – Logging and interpreting user interactions in information search and retrieval

Georg Buscher
DFKI GmbH
georg.buscher@dfki.de

Jacek Gwizdka
Rutgers University
jacekg@rutgers.edu

Jaime Teevan
Microsoft Research
teevan@microsoft.com

Nicholas J. Belkin
Rutgers University
belkin@rutgers.edu

Ralf Bierig
Rutgers University
bierig@rci.rutgers.edu

Ludger van Elst
DFKI GmbH
elst@dfki.uni-kl.de

Joeemon Jose
Glasgow University
jj@dcs.gla.ac.uk

1 Introduction

Modern information search systems can benefit greatly from using additional information about the user and the user's behavior, and research in this area is active and growing. Feedback data based on direct interaction (e.g., clicks, scrolling, etc.) as well as on user profiles/preferences has been proven valuable for personalizing the search process, e.g., from how queries are understood to how relevance is assessed. New technology has made it inexpensive and easy to collect more feedback data and more different types of data (e.g., gaze, emotional, or biometric data).

The workshop “Understanding the User – Logging and interpreting user interactions in information search and retrieval” was held in conjunction with the 32nd Annual International ACM SIGIR Conference. It focused on discussing and identifying most promising research directions with respect to logging, interpreting, integrating, and using feedback data. The workshop aimed at bringing together researchers especially from the domains of IR and human-computer interaction interested in the collection, interpretation, and application of user behavior logging for search.

WWW WORKSHOP REPORT

Query Log Analysis: Social and Technological Challenges

G. Craig Murray

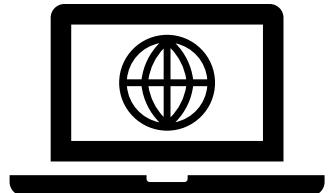
College of Information Studies
University of Maryland
graigm@umd.edu

Jaime Teevan

Microsoft Research
teevan@microsoft.com

Abstract

Analysis of search engine query logs is an important tool for developers and researchers. However, the potentially personal content of query logs raises a number of questions about the use of that data. Privacy advocates are concerned



Information Re-Retrieval: Repeat Queries in Yahoo's Logs

Jaime Teevan
MIT, CSAIL
Cambridge, MA 02138 USA
teevan@csail.mit.edu

Eytan Adar
University of Washington, CSE
Seattle, WA 98195 USA
eadar@u.washington.edu

Rosie Jones
Yahoo! Research
Burbank, CA 91504 USA
jonesr@yahoo-inc.com

Michael A. S. Potts
Yahoo! Research
Sunnyvale, CA 94089 USA
mpotts@yahoo-inc.com

ABSTRACT

People often repeat Web searches, both to find new information on topics they have previously explored and to re-find information they have seen in the past. The query associated with a repeat search may differ from the initial query but can nonetheless lead to clicks on the same results. This paper explores repeat search behavior through the analysis of a one-year Web query log of 114 anonymous users and a separate controlled survey of an additional 119 volunteers. Our study demonstrates that as many as 40% of all queries are re-finding queries. Re-finding appears to be an important behavior for search engines to explicitly support, and we explore how this can be done. We demonstrate that changes to search engine results can hinder re-finding, and provide a way to automatically detect repeat searches and predict repeat clicks.

Categories and Subject Descriptors

H3.3 [Information storage and retrieval]: Search and retrieval

General Terms

Measurement, Experimentation, Human Factors.

Keywords

Query log analysis, Web search, re-finding, repeat queries.

common and provides a detailed characterization of them. Given the pervasiveness of re-finding queries, we explore which search engine features support or hinder re-finding. In particular, we concentrate on changes in rank and demonstrate the detrimental impact of rank changes on this type of task. Making use of our understanding of re-finding behavior, we describe algorithmic methods to detect re-finding intent and suggest ways in which search engines can better support this behavior.

Log studies like the one presented here are valuable because they give a large-scale, realistic picture of users' actions. However, they give no insight into underlying motivation. To study re-finding through log analysis, it was necessary to try to glean from the data those queries which were intended to re-find information rather than find new information. Re-finding intent was approximated by looking for repeated clicks on the same search result in response to queries issued by the same user at different times. The query used to re-find the result may or may not be the same as the query used to find it originally. For example, if a person searched with the query “KPCC Southern California Public Radio” and clicked on the result <http://www.scp.org>, and then later clicked on the same result while searching for “scpr”, the behavior was considered re-finding. Because of our limited ability to automatically distinguish re-finding from finding

Session-based Information Retrieval

We can put information retrieval tasks in **context** based on

- the user's **short-term history** (Bennett et al., 2012);
- the user's long-term history (Keenoy and Levene, 2003);
- the user's situation (Zamani et al., 2017).



Short-term history is often formulated by the user interactions with the search engine in a short period of time (*e.g.*, a few minutes), referred to as a *search session*. Sessions are different from conversations in that one can pick up and continue a past conversation, while this is not possible in sessions.

Session-based Information Retrieval

Interactions in a session include:

- past queries
- retrieved documents
- clicked documents and other forms of implicit feedback

sessions are complex and they are not all alike, and they should be treated differently.

Basic Models for Session-based IR

- Shen et al. (2005) extended the language modeling framework by linearly interpolating the query language model with the context language model.
- Bennett et al. (2012) defined a wide range of session-related features to be used in a learning to rank model.

Session-based IR Evaluation

- TREC Session Track [Cartrette et al., 2016]
- sDCG [Järvelin et al. 2008]: a weighted sum of nDCG values for all queries in a session. Earlier queries get higher weights.
- nsDCG [Kanoulas et al. 2011]: a normalized version of sDCG
- These metrics have significant yet weak correlation with user's opinion. On the other hand, user's opinions are highly correlated with the performance of the worst and the last queries in the session [Jiang and Allan, 2016].

Exploratory Search

- Exploratory search is an information retrieval task in which the user is unfamiliar with the search task, unsure about the goal, or even unsure about how to complete the task.
- Exploratory search refers to a broad set of real-world search tasks that involve learning, investigation, planning, discovery, aggregation, and synthesis (Marchionini, 2006).

Exploratory Search and CIS



Natural language conversation is a convenient way for exploratory search tasks. In many exploratory search tasks, users experience difficulties describing their information needs using accurate keyword queries. This is mainly due to a misconception of the topics and/or the document collection. Information seeking conversations would be the natural solution for this problem as natural language conversation is perhaps the most convenient way of human communication and users can express their exploratory search needs quite easily.

Note: many conversations in the TREC CAsT Tracks (Dalton et al., 2019-2021) are basically addressing exploratory information seeking through conversation.

Formal Models for IIR

Some examples:

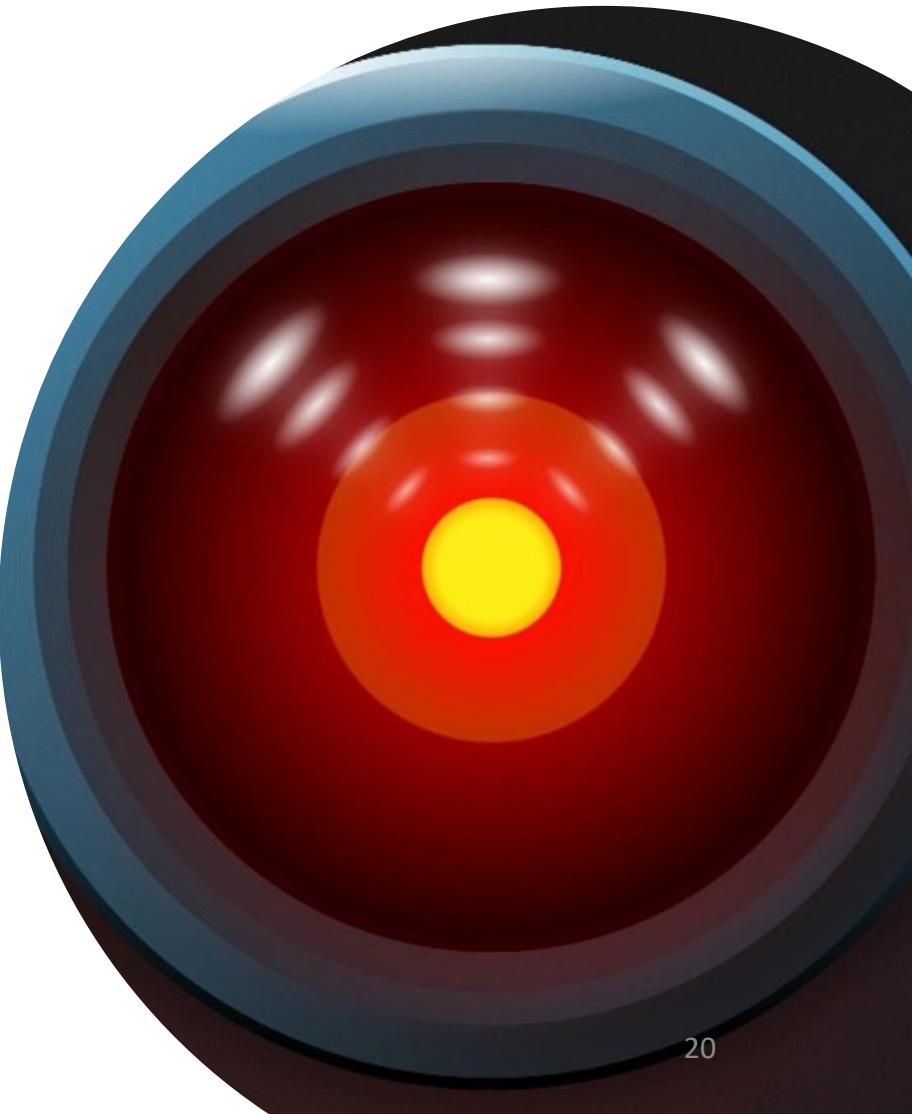
- Probability Ranking Principle for IIR [Fuhr, 2008]
- Economic Models for IIR [Azzopardi, 2011]
- Interface card model for IIR [Zhang and Zhai, 2015]

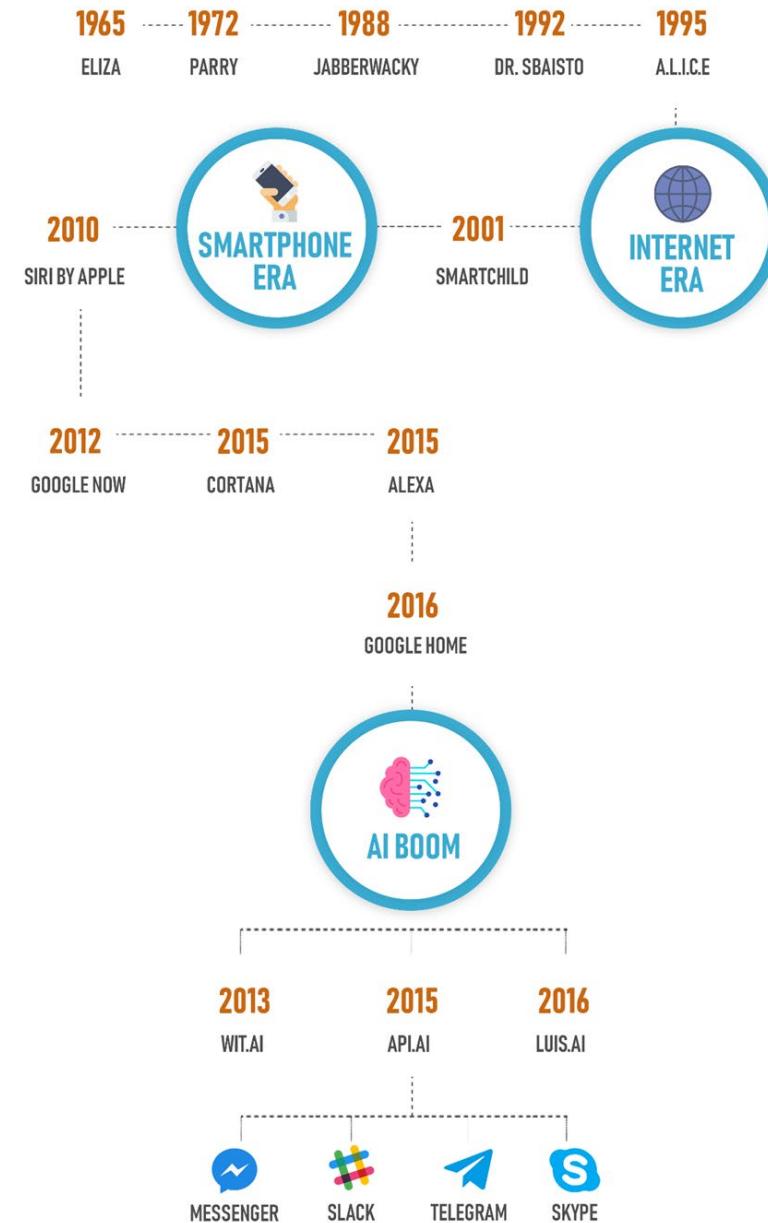
More reading:

- the comprehensive IIR survey by Kelly (2009)
- the recent SIGIR tutorial on IIR by Zhai (2020)



CIS also has roots in early dialogue systems research!







The Turing Test

Can a machine communicate
in a way indistinguishable
from humans?

The Loebner Prize

- Named after its founder and philanthropist Hugh Loebner
- The Loebner Prize is an annual contest based on the Turing Test and has run continuously since 1991.
- A Gold Medal (pictured) and \$100,000 prize is awarded to a machine that generates responses that are indistinguishable from a human.

The Focus of Dialogue Research

- **Chit-chat conversations:** mimicking human conversations mostly for entertainment
- **Task-oriented (goal-oriented) conversations:** helping users accomplish a task, e.g., hotel reservation

ELIZA (1964 – 1966)



- A pioneer chatbot created at MIT by Joseph Weizenbaum
- Named after Eliza Doolittle, a character in the satirical comedy *Pygmalion*
 - Her character represents a middle class girl learning speech patterns of the British upper class (learned to imitate the context she was given)
- The chatbot works in asking open questions based on the previous message it has received
 - Takes the persona of a Rogerian psychotherapist (someone who sits back and lets the other individual form the ideas and build up the conversation)

Welcome to

| | | | | |
|--------|-------|-----|-------|---------|
| EEEEEE | LL | III | ZZZZZ | AAAAA |
| EE | LL | II | ZZ | AA AA |
| EEEEEE | LL | II | ZZZ | AAAAAAA |
| EE | LL | II | ZZ | AA AA |
| EEEEEE | LLLLL | III | ZZZZZ | AA AA |

Eliza is a mock Rogerian psychotherapist.

The original program was described by Joseph Weizenbaum in 1966.

This implementation by Norbert Landsteiner 2005.



ELIZA: Is something troubling you ?

YOU: Men are all alike.

ELIZA: What is the connection, do you suppose ?

YOU: They're always bugging us about something or other.

ELIZA: Can you think of a specific example ?

YOU: Well, my boyfriend made me come here.

ELIZA: Is it important to you that your boyfriend made you come here ?

YOU: He says I'm depressed much of the time.

ELIZA: I am sorry to hear that you are depressed.

YOU: It's true. I am unhappy.

ELIZA: Can you explain what made you unhappy ?

YOU:

Interesting observation: Users began to confide in ELIZA (which troubled Weizenbaum), sharing their deep thoughts and feelings with the bot.

PARRY (1972)



- PARRY creates ‘his’ own messages without necessarily depending purely on prior context like ELIZA.
- Introduced by Kenneth Mark Colby, a psychiatrist and computer scientist, a separate take of diverting the Chit-Chat from itself
- PARRY takes the persona of a paranoid patient with schizophrenia
 - Using assumptions and “emotional responses” based on user input, PARRY gives responses that are erratic in manner and loosely based on the input
 - Instead of creating questions based on previous responses, it ‘provokes controversies’ to engage with the user
- A tool to teach psychiatrists to communicate with paranoid schizophrenic diagnoses

ELIZA vs. PARRY

- In a conference in 1972, the schizophrenic patient PARRY talks to a ELIZA, the psychiatrist bot.
- It goes back and forth in a manner like this, with ELIZA questioning PARRY on its erratic responses



Jabberwacky (1991)

- Used more of a humorous approach to chit-chatting
- Later in its use was used for more marketing and sales with its entertaining responses
 - 2003, 3rd place in Loebner Prize
 - 2004, 2nd place in Loebner Prize
- Eventually replaced by a new variant known as Cleverbot (2008)

Jabberwacky: Have we spoken before?

User: I want to start a campfire!

Jabberwacky: How do you plan to do that?

User: Got a match?

Jabberwacky: Got an apple here.

User: I really must say, I love a red, yellow or even a green apple. They taste great.

Jabberwacky: But I thought you were going to flirt with me!

User: It all makes sense.

Jabberwacky: Perhaps it makes sense to you, but it does not make sense to me.

User: Well, I guess that changes everything.

Jabberwacky: It should.

User: If I could give you gentle peace, love and joy, that never ceased, I would.

GUS: Genial Understander System (1977)

- Designed for travel planning, later expanded to several domains
- **Frame-based** modular rule-based system representing user intentions
- ***Frame*:** Representation of the current user intention and a collection of slots.
 - A set of frames are called a **domain ontology**
- Some GUS aspects are still used in many production systems such as Siri and Alexa.
- Extension of GUS, more commonly used in research systems

GUS (1977)

| | |
|---------|---|
| GUS: | Hello. My name is GUS. I can help you plan a simple trip by air. Where do you want to go ? |
| Client: | I want to go to San Diego on May 28 |
| GUS: | What time do you want to leave? |
| Client: | I must be in San Diego before 10 am |
| GUS: | Would you like P.S.A. flight 102 that arrives at 9.15 am ? |
| Client: | What is the next flight ? |
| GUS: | Air California flight 310 that leaves at 8.30am and arrives at 10.00am |
| Client: | I'll take the first one |
| GUS: | In what name should I make a reservation, please? |
| Client: | Dan Bobrow |
| GUS: | I have confirmed the following flight: P.S.A. flight 102 on Wednesday May 28 from San Jose to San Diego that leaves at 7.30 am and arrives at 9.15 am. |
| GUS: | What date do you want to return on ? |
| Client: | On Friday in the evening. |
| GUS: | Would you like the flight that leaves at 7.45 pm ? |
| Client: | That's fine. |
| GUS: | I have confirmed the following flight: P.S.A. flight 307 on Friday May 30 from San Diego to San Jose that leaves at 7.45 pm and arrives at 9.30 pm Thank you for calling. Goodbye |

GUS - Control Structure

- The system's **goal is to fill slots in frames by understanding user intent**

*User: I need to book a flight **from Boston to Seattle landing at 6.00 p.m. next Thursday.***

| Slot | Type | Question Template |
|------------------|------|--------------------------------------|
| ORIGIN CITY | city | “From what city are you leaving?” |
| DESTINATION CITY | city | “Where are you going?” |
| DEPARTURE TIME | time | “When would you like to leave?” |
| DEPARTURE DATE | date | “What day would you like to leave?” |
| ARRIVAL TIME | time | “When do you want to arrive?” |
| ARRIVAL DATE | date | “What day would you like to arrive?” |

- System detects the slot to be filled and conducts database query
- After slot and frame, switch dialog control to new frame
- System may ask user questions based on pre-defined templates or rules

GUS - NLU for slot filling

Three main goals:

- **Domain Classification** - Flight Booking, Movie Recommendation, etc.
- **Intent Detection** - Maps user utterance to Intent Frame

Wake me tomorrow at 6

should give an intent like this:

DOMAIN: ALARM-CLOCK
INTENT: SET-ALARM
TIME: 2017-07-01 0600-0800

- **Slot Filling** - Maps user utterance to slots in a frame using grammar rules

Phoenix System (1994)

- Human designed semantic grammars

SHOW

→ show me | i want | can i see|...

DEPART_TIME_RANGE

→ (after|around|before) HOUR |
morning | afternoon | evening

HOUR

→ one|two|three|four...|twelve (AMPM)

FLIGHTS

→ (a) flight | flights

AMPM

→ am | pm

ORIGIN

→ from CITY

DESTINATION

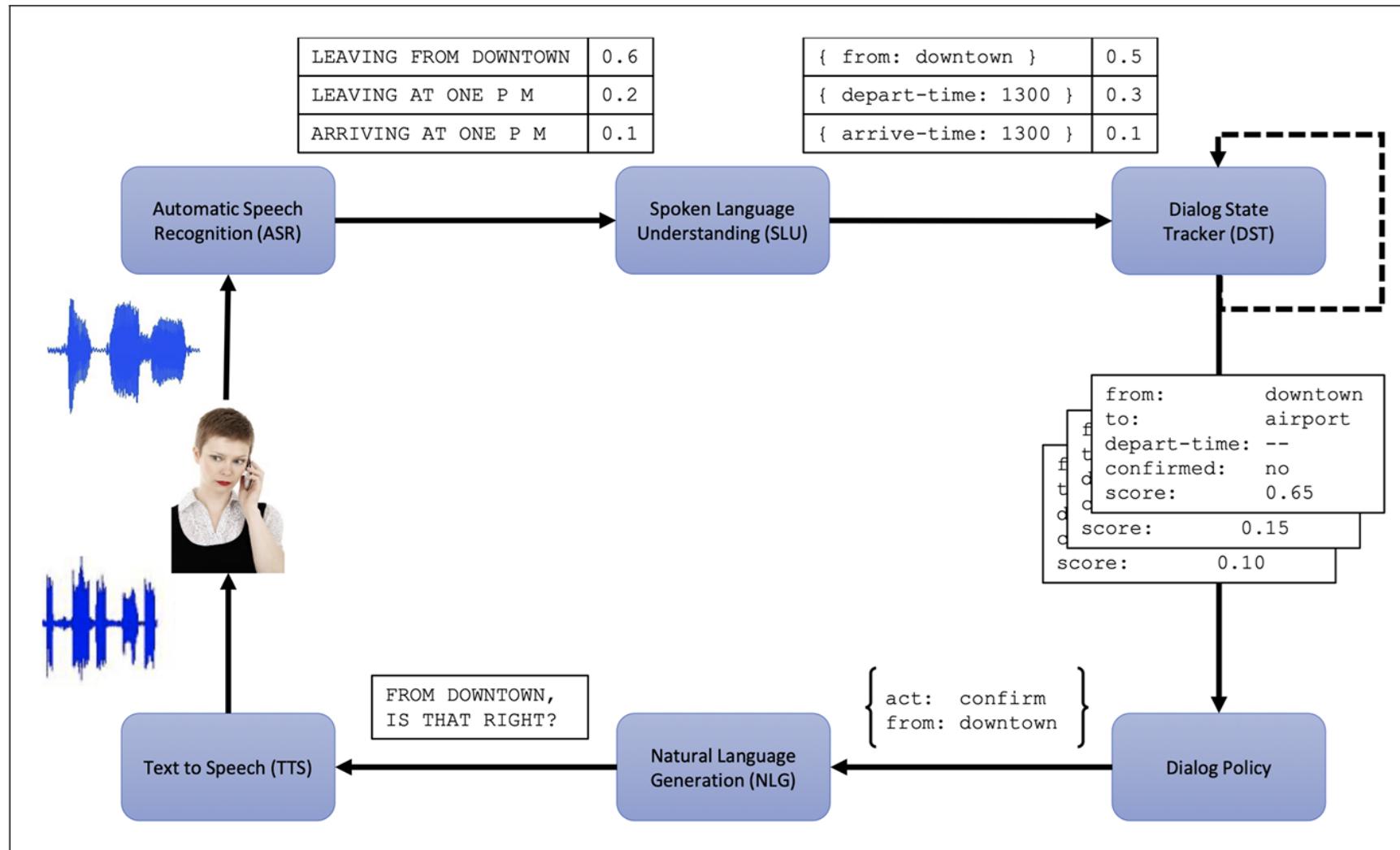
→ to CITY

CITY

→ Boston | San Francisco | Denver | Washington

- Speech acts are assigned to sequences

Dialog State Architecture



Hidden Information State System (2010)

- HIS uses a version of Markov Decision Process to map an Utterance to a Dialogue Act

- Dialogue Act:** A grounded speech act

| Tag | Sys | User | Description |
|----------------------------------|-----|------|--|
| HELLO($a = x, b = y, \dots$) | ✓ | ✓ | Open a dialogue and give info $a = x, b = y, \dots$ |
| INFORM($a = x, b = y, \dots$) | ✓ | ✓ | Give info $a = x, b = y, \dots$ |
| REQUEST($a, b = x, \dots$) | ✓ | ✓ | Request value for a given $b = x, \dots$ |
| REQALTS($a = x, \dots$) | ✗ | ✓ | Request alternative with $a = x, \dots$ |
| CONFIRM($a = x, b = y, \dots$) | ✓ | ✓ | Explicitly confirm $a = x, b = y, \dots$ |
| CONFREQ($a = x, \dots, d$) | ✓ | ✗ | Implicitly confirm $a = x, \dots$ and request value of d |
| SELECT($a = x, a = y$) | ✓ | ✗ | Implicitly confirm $a = x, \dots$ and request value of d |
| AFFIRM($a = x, b = y, \dots$) | ✓ | ✓ | Affirm and give further info $a = x, b = y, \dots$ |
| NEGATE($a = x$) | ✗ | ✓ | Negate and give corrected value $a = x$ |
| DENY($a = x$) | ✗ | ✓ | Deny that $a = x$ |
| BYE() | ✓ | ✓ | Close a dialogue |

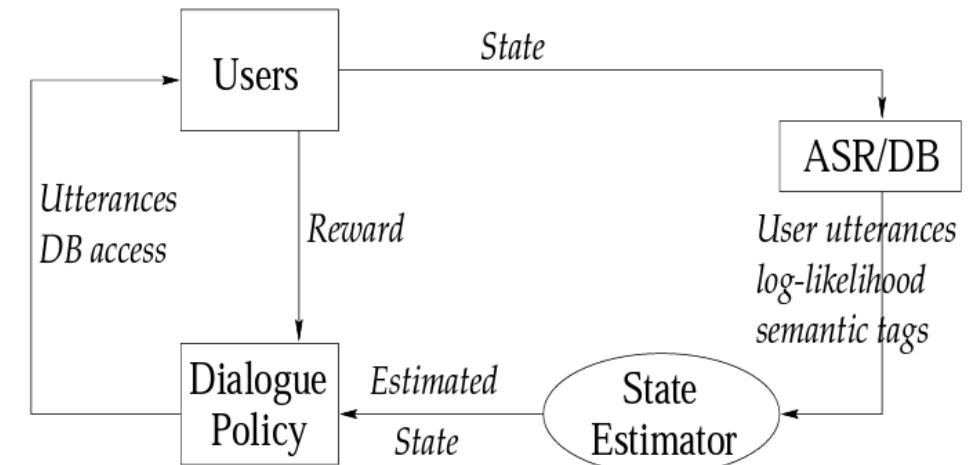
| Utterance | Dialogue act |
|---|---|
| U: Hi, I am looking for somewhere to eat. | hello(task = find, type=restaurant) |
| S: You are looking for a restaurant. What type of food do you like? | confreq(type = restaurant, food) |
| U: I'd like an Italian somewhere near the museum. | inform(food = Italian, near=museum) |
| S: Roma is a nice Italian restaurant near the museum. | inform(name = "Roma", type = restaurant, food = Italian, near = museum) |
| U: Is it reasonably priced? | confirm(pricerange = moderate) |
| S: Yes, Roma is in the moderate price range. | affirm(name = "Roma", pricerange = moderate) |
| U: What is the phone number? | request(phone) |
| S: The number of Roma is 385456. | inform(name = "Roma", phone = "385456") |
| U: Ok, thank you goodbye. | bye() |

Dialog Policy

- In rule based systems, dialog policies are hand-designed
- More modern systems use **reinforcement learning**

NJFun System (2002):

- Provides information about fun things to do in New Jersey
- One of the earliest to use RL



Conclusions

- CIS has roots in both early IR and Dialogue research.
- From the IR perspective:
 - Interactive IR (IIR)
 - Formal models of IIR
 - Session-based IR
 - Exploratory IR
- From the Dialogue perspective:
 - Chat-chat conversations
 - Task-oriented dialogues
 - Dialogue understanding and generation

Questions



Foundations and Trends in
Information Retrieval
(to appear)

Conversational Information Seeking

Hamed Zamani, Johanne Trippas, Jeff Dalton,
and Filip Radlinski



#TheCISTutorial

<https://arxiv.org/pdf/2201.08808.pdf>