Tutorial on User Simulation for Evaluating Information Access Systems

Krisztian Balog & ChengXiang Zhai

November 25, 2023

Krisztian Balog
University of Stavanger
Norway

ChengXiang Zhai
University of Illinois at Urbana-Champaign
USA

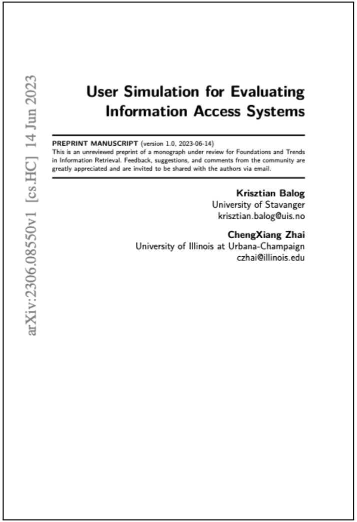
Overview

Introduction and Background

Supplementary Materials

Tutorial is based on a book that is currently under review at Foundations and Trends in Information Retrieval.

- Preprint: `https://arxiv.org/abs/2306.08550`
- Website: `https://usersim.ai`



User Simulation for Evaluating Information Access Systems

PREPRINT MANUSCRIPT (version 1.0, 2023-06-14)
This is an unreviewed preprint of a monograph under review for Foundations and Trends in Information Retrieval. Feedback, suggestions, and comments from the community are greatly appreciated and are invited to be shared with the authors via email.

Krisztian Balog
University of Stavanger
krisztian.balog@uis.no

ChengXiang Zhai
University of Illinois at Urbana-Champaign
czhai@illinois.edu

Information Access

- Information access systems aim to **help users find information**
  - Search engines, recommender systems, and conversational assistants
  - "Access to the right information at the right time"
- Interactions with these systems generally involve
  - entering **information needs** or preferences (e.g., typing queries, rating items, or asking natural language questions)
  - interacting with **information objects** (e.g., by clicking, typing, or speaking)
  - that are presented by the system on some device (e.g., desktop, tablet, smart phone, or smart speaker))
  - in some modality or combination of modalities (e.g,. text, rich snippets, voice)
- The **evaluation** of these systems represents an **open challenge**

Information Access Tasks

- *Pull mode*: user takes the initiative and uses a search engine to find information
- *Push mode*: the system takes the initiative and recommends relevant information to the user
- Search and recommendation are "two sides of the same coin" and involve:
  - Modeling a user's information need and preferences
  - Matching an information object with a user's interest
  - Ranking items accurately
  - Learning from user feedback
  - Evaluating a ranked list to assess its utility to a user
- *Mixed initiative*: conversational assistants facilitate both search and recommendation via natural language interactions

Evaluation Goals

- Two main evaluation goals are distinguished:
  - To know the **actual utility of technology**
    - How useful is a given system to a group of users?
    - Needs to be answered with an interpretable absolute value to quantify the performance and utility of the system.
    - To be used to inform decisions like: "Is it worth the investment?" or "Is it worth deploying?"
  - To establish a **relative comparison between two systems**
    - Required in order to make progress in developing better systems and advance research
    - Weaker requirement that only needs a relative measure that is correlated with the absolute difference

Evaluation Methodologies

- Reusable test collections
  - Standard evaluation methodology for making relative comparisons between two systems in a repeatable and reproducible manner
  - Limited ability to capture many aspects of users and interactions adequately; the user is abstracted away
- User studies
  - Provides the highest fidelity in terms of capturing real users' interactions with an actual system in a controlled setting
  - Costly to run, not reproducible
- Online evaluation
  - Observing real users of a fully operational system and assessing the system's performance by analyzing the recorded user behaviour
  - Enables measuring the actual utility of a system; scalable
  - Not reproducible, no control over users

Challenges and Simulation-based Evaluation

- None of the previous methodologies enable comparison of multiple interactive information access systems using reproducible experiments
  - Test collection-based evaluation is static in nature
  - Lack of reproducibility when real users are involved
- It is important to evaluate the *overall effectiveness* of a system
  - Commonly, complex tasks are decomposed into a series of smaller and simpler components
  - These can be abstracted, studied and addressed in isolation (using reusable test collections)
  - However, the evaluation of individual components alone is insufficient
  - The ultimate goal is to evaluate the *whole* system from a user's perspective
- The evaluation of an interactive system's overall effectiveness must involve a user in some way
  - The involvement of real users inherently leads to non-reproducible experiments
  - Simulated users can be controlled and thus enable reproducible experiments

User Simulation

- Informal definition: having an intelligent agent to simulate how a user interacts with a system
- User simulation has many uses, including
  - Performing **large-scale automatic evaluation** of interactive systems (i.e., without the involvement of real users)
  - Gaining **insight into user behaviour** to inform the design of systems and evaluation measures
  - **Analyzing system performance** under various conditions and user behaviours (answering what-if questions, such as "What is the influence of X on Y?")
  - **Generating synthetic data** with the purpose of training machine learning models, especially reinforcement learning
- For relative comparisons of systems, simulation does not need to be perfect; it is enough to identify relative system differences

User Simulation

- It is assumed that there is some information available about
  - the **system** and its **user interface** (e.g., search engine with a query box and navigable search result lists)
  - the **user's task** (e.g., collecting as many relevant information items as possible or finding a suitable product to purchase)
  - the **user** (e.g., background knowledge, context)
- Goal: simulate all the actions a user can potentially take when interacting with the system to perform some task given any particular interaction context

User Simulation

- Formally/computationally/mathematically define a user in the context of finishing a task using an interactive system, including particularly specifying how the user would behave in each interaction context/scenario
- Configuration variables for user simulation:
    1. Task (T): a user's behaviour varies according the task
    2. System (S): a user's behaviour depends on the system (functions) that the user interacts with
    3. User information (U): different users may behave differently when finishing the same task using the same system
- As a computation problem: Given T, S, U, create an agent to simulate every action that user U may take when finishing T by using system S

Simulation Approaches

- Two broad approaches:
  - *Model-based*: can be rule-based (based on knowledge about how users behave) or interpretable probablistic models (parameters set heuristically or estimated based on observed user data)
  - *Data-driven*: maximize accuracy of fitting any observed real user data, without necessarily imposing interpretability (supervised ML)
- Accurate simulation of observable behaviour may require simulation of latent behaviour (e.g., cognitive state of a user), which makes simulation more interpretable (via interpretable generative models)
- Interpretability is desirable to enable the testing of verifiable hypotheses about users and ensure that evaluation results are meaningful
  - Varying the parameters corresponds to the simulation of different kind of users

Partial vs. Complete User Simulation

- Simulation of an action of a user: Given an interaction context (system environment), predict what action a user would take (e.g., given a snippet in a list of search results, predict whether a user would click on it)
- Simulation of a sequence of actions of a user: Given an interaction context, predict the whole sequence of multiple actions that a user would take (need to consider dependency between actions)
- Simulation of a user's interactions in a whole session of finishing a task (there may be multiple sequences of interactions)
- Simulation of a user's general preferences and behaviour across tasks

Overview of User Simulation

Background

- Information retrieval
  - Interactive IR
  - Recommender systems
  - Conversational search and recommendation
- Dialogue systems
- User modeling

Background / IR & RecSys

Both search and recommendation address the problem of providing users with items that are estimated to be relevant to the user's information need, preferences, and/or context, often presented as a ranked list

- Early simulation work in IR
  - Synthetic queries and documents to analyze the effect of changes in query characteristics on the number of documents retrieved (Cooper, 1973)
  - Effectiveness of relevance feedback (Spärck Jones, 1979; Harman, 1992)
- "Second wave" with Interactive IR in the 2000s
  - Relevance feedback (Leuski, 2000; Keskustalo et al., 2008)
  - Query generation (Azzopardi and de Rijke, 2006; Baskaya et al., 2012)
  - Scanning/examination/stopping behaviour (Turpin et al., 2009; Baskaya et al., 2013; Maxwell et al., 2015)

Background / Interactive IR

While IR tends to have a strong system focus, interactive information retrieval (IIR) focuses more on users and how they interact with the retrieval system

- Early studies pointing out user effort as an important factor (Cleverdon and Kean, 1968; Salton, 1970)
- Early IIR measures can be categorized around relevance, efficiency, utility, user satisfaction, and success (Su, 1992)
- Important research finding: discrepancy between interactive and non-interactive evaluation results
  - No significant relationship between the effectiveness of a search engine, measured by Mean Average Precision, and real user success in a precision-oriented task (Turpin and Scholer, 2006)
  - Users can adapt their behaviour and can be just as successful with a degraded search system than with a standard one (Smith and Kantor, 2008)

Background / Dialogue Systems

The goal of task-based dialogue systems is to help the user accomplish some task, such as make a restaurant reservation or buy a product

- Important idea: modeling human-computer dialogue formally as a Markov Decision Process (MDP) (Levin et al., 2000; Young, 1999)
- Simulation has become the predominant form of dialogue policy learning (Schatzmann et al., 2006; Young et al., 2010)
- Using simulation for evaluation is much less studied

Background / User Modeling

User simulation can be regarded as developing a complete and operational user model

- Descriptive vs. formal models
  - *Descriptive models* can provide reasoning and (post-hoc) explanation behind user behaviour
  - *Formal models* are expressed mathematically and have predictive power about why users behave in a certain way

Summary

- Most work has been done on simulating users of search engines
  - Formulating queries
  - Examining search results
  - Modeling search strategies
  - Variation of user behavior
- Much less work has been done on simulating users of recommender systems
  - Possible reasons: 1) No standard user interface for recommender systems; 2) Research is more focused on improving the recommender algorithms
  - Most work so far is on click modeling/prediction, often for the purpose of optimizing recommendation accuracy, instead of accurately modeling real users
- Growth of work on simulating users of conversational assistants (may be pushed by the necessity of evaluating systems)

Simulation-based Evaluation Frameworks

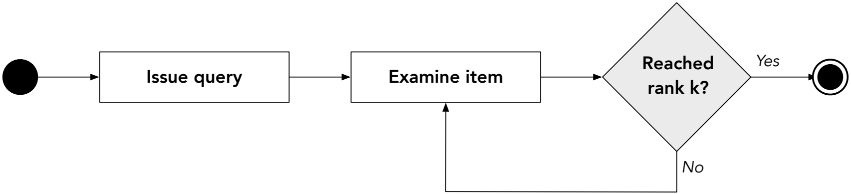
Outline

- Traditional evaluation measures and user simulation
- Limitations of traditional evaluation measures
- A general simulation-based evaluation framework
- Traditional evaluation measures as special cases
- Beyond search list evaluation: Simulation-based evaluation of interactive search interfaces

Traditional (Test Collection-based) Evaluation

- Components of an IR test collection
  - Collection of documents
  - A set of queries
  - Corresponding relevance judgments
- System is run to generate retrieval results for each query
- Retrieval performance is measured for each query using various evaluation metrics (e.g., Precision, Recall, NDCG) $\Rightarrow$ perceived utility of a result list from the user's perspective

Traditional Evaluation Measures as Naive User Simulators



- User model: Sequentially browse the ranked list of results up to rank position k and examine each item
- E.g., Precision@k, Recall@k, MAP

Measures based on Explicit Models of User Behaviour

Virtually all measures attempt to quantify the performance of a search result based on a combination of four factors:

- The assumed **user task** (e.g., high precision vs. high recall)
- The assumed **user behaviour** when interacting with the results
- Measurement of the **reward** a user would receive from examining the result
  - Early IR measures defined reward based on relevance-based gains
  - Later, novelty and diversity of the search results were also considered
- Measurement of the **effort** a user would need to make in order to receive the reward
  - Uniform vs. longer documents would take more effort/time

Measures based on Explicit Models of User Behaviour

Virtually all measures attempt to quantify the performance of a search result based on a combination of four factors:

- The assumed **user task** (e.g., high precision vs. high recall)
- The assumed **user behaviour** when interacting with the results
- Measurement of the **reward** a user would receive from examining the result
  - Early IR measures defined reward based on relevance-based gains
  - Later, novelty and diversity of the search results were also considered
- Measurement of the **effort** a user would need to make in order to receive the reward
  - Uniform vs. longer documents would take more effort/time

Limited to evaluating a ranked list of results; insufficient in highly interactive settings

A Step Toward Capturing Interaction: Session-based Measures

With the assumption of a ranked list of results, query-based measures can be generalized to create session-based measures.

- **Session nDCG (sDCG) measure** (Järvelin et al., 2008): Concatenate all the search results in a session to form a single ranked list of documents, and then apply nDCG $\Rightarrow$ more discounting on results returned in later in a session
- **Expected Global Utility over a session** (Yang and Lad, 2009): Model the uncertainty of a user's browsing behaviour and compute the expected utility w.r.t. the distribution of all possible user browsing behaviours
- **Modeling a user's browsing behaviour in a session as a "path"** (Kanoulas et al., 2011): Capture the perceived ranking of all the documents a user has interacted with in a session as a single ranked list; any measure can then be defined based on such a perceived ranked list for the whole session

A Step Toward Capturing Interaction: Session-based Measures

With the assumption of a ranked list of results, query-based measures can be generalized to create session-based measures.

- **Session nDCG (sDCG) measure** (Järvelin et al., 2008): Concatenate all the search results in a session to form a single ranked list of documents, and then apply nDCG ⇒ more discounting on results returned in later in a session
- **Expected Global Utility over a session** (Yang and Lad, 2009): Model the uncertainty of a user's browsing behaviour and compute the expected utility w.r.t. the distribution of all possible user browsing behaviours
- **Modeling a user's browsing behaviour in a session as a "path"** (Kanoulas et al., 2011): Capture the perceived ranking of all the documents a user has interacted with in a session as a single ranked list; any measure can then be defined based on such a perceived ranked list for the whole session

Still limited to evaluating a ranked list of results ⇒ Can we evaluate more sophisticated interactions?

A General Simulation-based Evaluation Methodology

- A collection of user simulators are constructed to approximate real users
- A collection of task simulators are constructed to approximate real tasks
- Both user simulators and task simulators can be parameterized to enable modeling of variation in users and tasks
- Evaluation of a system
  - Have a simulated user perform a simulated task by using (interacting with) the system
  - Compute various measures based on the entire interaction history of the whole "task session"

A General Formal Framework for Simulation-based Evaluation (Zhang et al., 2017)

- Let S be a system, U be a user, and I be the whole process of the interaction of U and S to finish task T
- Measure the system's performance based on I. From a user's perspective, we can measure the performance in two dimensions:
  - Interaction Reward, $R(I, T, U, S)$: the total reward the user has received via the interaction
  - Interaction Cost, $C(I, T, U, S)$: the total cost of the interaction
- In general, the more interaction actions the user makes, the more reward the user can potentially receive and the more cost the user would have to bear (since the user needs to make more effort)
- If one single measure is needed, the reward and cost can be combined, which can be in many different forms

Consideration of Stochastic User Actions

- When the user U is a simulated user, the interaction sequence I may be uncertain or stochastic
- In such a case, a more general measure of reward or cost can be defined as the expected Interaction Reward or Interaction Cost w.r.t. the distribution of all the possible interaction sequences that the simulated user U may make with system S, i.e., $P(I|T,U,S)$
- Expected Simulator Reward: $R(T,U,S) = \sum_I P(I|T,U,S)R(I,T,U,S)$
- Expected Simulator Cost: $C(T,U,S) = \sum_I P(I|T,U,S)C(I,T,U,S)$

Refinement

- Assumption: I is a sequence of specific user actions taken in response to a sequence of Interface Card, generated by system S
- Refinement: Reward and cost of an interaction sequence can be further defined based on the reward and cost of an individual action
- Refined Formalization of Interaction Action: (z, a, q) (Zhang and Zhai, 2015)
  - q: an interface card (i.e., a dynamic user interface) generated by the system
  - z: a representation of the user's state during the interaction
  - a: an action taken by the user in response to the interface card q
- Refined formalization of an interaction sequence:
  $I = ((z_1, a_1, q_1), (z_2, a_2, q_2), ..., (z_n, a_n, q_n))$

Action-Level Reward and Cost

- Action-Level Refinement of Reward and Cost

$$R^t(I,T,U,S) = \sum_{i=1}^{t} r(a^i|z^i, q^{i-1})$$

$$C^t(I,T,U,S) = \sum_{i=1}^{t} c(a^i|z^i, q^{i-1})$$

- How to combine the reward and cost measures is application specific (e.g., both reward and cost can be potentially weighted based on status of task completion)
- The distributions of reward and cost across all interaction sequences are also meaningful (e.g., it might make sense to minimize the worst cost)

Classic IR Simulator

- Task: find (all) relevant documents
- Interface card: document (snippet)
- User simulator
  - User actions: click, skip (and read next), or stop
  - User always clicks a relevant document when encountering one
  - User always skips a non-relevant document when encountering one
  - User will stop when the effort/cost reaches a budget (or when the user finds the first relevant document in the case of Mean Reciprocal Rank)
- Lap reward: 1 (relevant doc); 0 (non-relevant doc) $\Rightarrow$ Cumulative reward: # relevant docs
- Lap cost: 1 (for scanning each doc/snippet) $\Rightarrow$ Cumulative cost: # docs scanned by the simulated user
- User state: cumulative reward and cost

Classic IR Metrics

- Precision: $R(I, T, U, S)/C(I, T, U, S)$
- Recall: $R(I, T, U, S)/N$, N = maximal possible reward
- Remarks
  - Assumes user stops when the list is exhausted
  - Precision@K and Recall@K: K = cost budget
  - Precision emphasizes more on cost
  - Recall emphasizes more on task completion

Average Precision

- Variable-recall simulator
  - Classical IR simulator with the task of finding N' relevant documents ($N' \in [1..N]$)
  - Stops and only stops when the task is finished
- Average Precision (AP)
  - Average $R(I, T, U, S)/C(I, T, U, S)$ across N variable-recall simulators with N' ranging from 1 to N respectively
  - AP@K: K $=$ cost budget

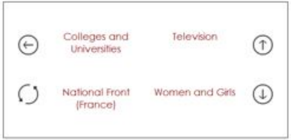
Application of Framework: Evaluating of Tag-based Search Interfaces

- Examples of an interactive search interface beyond ranking
  - Traditional interface: static layout
    - Medium screen: tag list alongside document list
    - Small screen: only tag list or document list at a time, and user needs to click "switch" to switch between the two lists
  - ICM interface: dynamic layout (Zhang et al., 2017)
  - Evaluation based on simulators
    - Task: find target document(s)
    - Simulator never stops until task is completed
    - Metrics: interaction cost

Tag-based Search Interfaces: Simulator Action Model

- If a target document is shown, user always clicks it
- Otherwise, if a tag related to a target document is shown, user always clicks it
- Otherwise:
  - On ICM: User always goes to "next page"
  - On medium static interface: user scrolls document list with probability τ, and scrolls tag list with probability (1- τ)
  - On small static interface:
    - If user is on document list, user scrolls list with probability τ_1 and switches list with probability (1- τ_1)
    - If user is on tag list, user scrolls list with probability τ_2 and switches list with probability (1- τ_2)

Sample Interfaces and User Actions



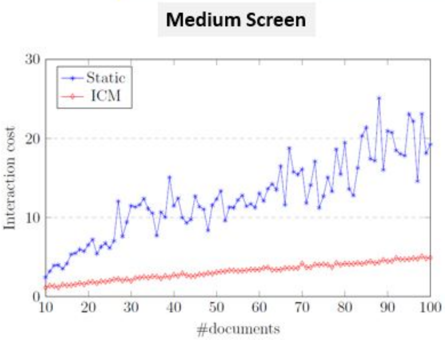
Simulator scrolls list with probability τ_2 and switches list with probability $(1- \tau_2)$

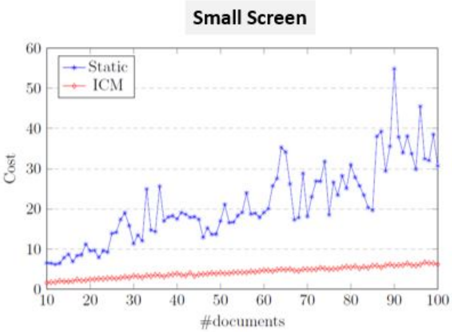
Simulator scrolls list with probability τ_1 and switches list with probability $(1- \tau_1)$

Simulator scrolls document list with probability τ, and scrolls tag list with probability $(1- \tau)$

Results of Simulation-based Evaluation

Interface Card Model has consistently lower interaction cost than the static interface





Validation from Real User Experiment

- Real user experiment (Zhang et al., 2017)
  - ICM is more efficient than static interface
  - The difference is higher on small screen than on medium screen
  - These results are consistent with results of simulation-based evaluation
- Insights about real user behavior
  - Users can well utilize the tag list on the medium screen, but cannot make full use of the tag list on the small screen

Screen size	Sample size	Workers' average
Small	42	$\hat{\tau}_1 = 0.845$, $\hat{\tau}_2 = 0.370$
Medium	38	$\hat{\tau} = 0.211$

Table 6.2: Real user action averages

Summary

- A general simulation-based evaluation framework is introduced
  - Evaluation is based on the expected reward and cost of a sequence of interactions between a user and a system
  - Sufficiently general to cover evaluating any interactive information access systems
  - Can be refined with different ways to define actions and action-level cost/reward and different ways to aggregate them
- Traditional evaluation measures can be interpreted as simulating naive users in the general simulation-based evaluation framework
- The framework enables meaningful simulation-based evaluation of interactive search interfaces/systems that go beyond ranking documents

User Simulation and Human Decision-making

User Simulation and Human Decision-making

- Cognitive Models
- Process Models
- Strategic Models
- Choice and Decision Making in Recommender Systems
- Mathematical Framework

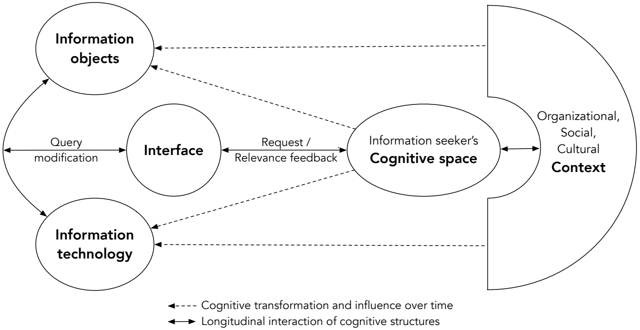
Cognitive Models

Focus on the **cognitive processes** underlying the information-seeking activity (individual's internal representation of a problem situation).

- Belkin's Anomalous State of Knowledge (ASK) hypothesis
  - *"An information need arises from a recognized anomaly in the user's state of knowledge concerning some topic or situation and that, in general, the user is unable to specify precisely what is needed to resolve that anomaly"* (Belkin et al., 1982)
  - Proposes a specific reason as to why people engage in an information-seeking behaviour
  - Assumes the presence of a human intermediary and proposes the ASK to be resolved via co-operative *dialogue* between the user and the intermediary

Cognitive Models

- Information seeking and retrieval (IS&R) research framework (Ingwersen and Järvelin, 2005)
  - Detailed description of essential processes from both the user and system perspectives
  - Emphasizes the *interaction* between the information seeker(s) and the environment surrounding that individual
  - Remains at a very high level of conceptualization



User Simulation and Human Decision-making

- Cognitive Models
- Process Models
- Strategic Models
- Choice and Decision Making in Recommender Systems
- Mathematical Framework

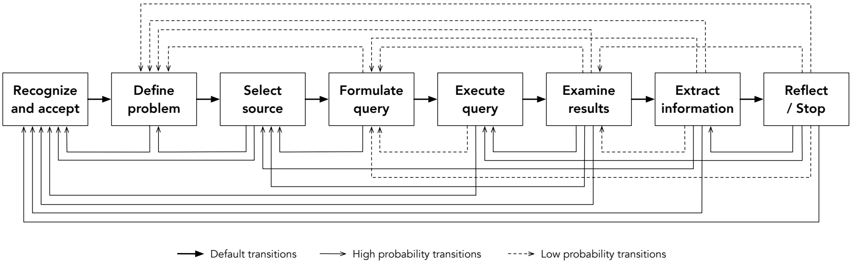
Process Models

Represent the **different stages and activities** during the search process.

- Kuhlthau (1991) identifies six stages:
  1. *Initiation*, recognizing a need for information
  2. *Selection* of the general topic and approach that is expected to yield the best outcome
  3. *Exploration* of the general topic in order to further personal understanding
  4. *Formulation*, where a focused perspective on the topic emerges
  5. *Collection* of the information related to the focused topic
  6. *Presentation*, which completes the search and prepares the results to be presented or used.
- These stages characterize complex information needs and are not necessarily representative for more light-weight tasks

Process Models

- Marchionini (1995) decompose information-seeking into eight sub-processes
  - Sub-processes do not necessarily follow each other in a sequential order, but may develop in parallel and at different rates
  - Sub-processes are further categorized into three classes: (1) understanding, (2) planning and execution, and (3) evaluation and use
    - (1) is mainly a mental activity,(2) and (3) are both mental and behavioural activities



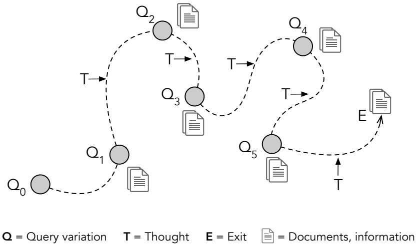
User Simulation and Human Decision-making

- Cognitive Models
- Process Models
- Strategic Models
- Choice and Decision Making in Recommender Systems
- Mathematical Framework

Strategic Models

Describe **tactics** (high level search strategies) that users employ when searching for information, using analogies from the physical world.

- *Berry-picking model* (Bates, 1989)
  - Considers information seeking analogous to foragers looking for food
  - It assumes that searchers' needs are not satisfied by a single set of retrieved results, scattered like berries on bushes
  - As searchers encounter new pieces of information along the way, those might give them new ideas and directions to follow
  - The model is supported by observational studies (O'Day and Jeffries, 1993; Borgman, 1996)



Strategic Models

- *Information foraging theory* (Pirolli and Card, 1999)
  - Applies ideas from optimal foraging theory $\Rightarrow$ the searcher maximizes the rate of gaining valuable information over time
    - Optimal foraging theory explains how animals maximize their fitness while they search for food (i.e., gain the most energy for the lowest cost)
  - *Patch* is an area where food can be acquired $\Rightarrow$ SERP
    - Foragers need to decide how long they want to stay in a patch before moving to the next patch $\Rightarrow$ examine SERP vs. issue a new query
  - *Scents* indicate to animals their chances of finding prey $\Rightarrow$ *information scent* are cues presented to on web pages or SERPs
    - When information scent starts to decrease, searchers transition to other information sources

User Simulation and Human Decision-making

- Cognitive Models
- Process Models
- Strategic Models
- Choice and Decision Making in Recommender Systems
- Mathematical Framework

Choice and Decision Making in Recommender Systems

The ASPECT model (Jameson et al., 2014) distinguishes six human *choice patterns*.

- *Attribute-based choice*: options can be described in terms of attributes, some of which are considered more important than others
- *Consequence-based choice*: consider the consequences of choosing a particular option
- *Experience-based choice*: the person has past experience either with the given choice situation or with particular options
- *Socially-based choice*: people often let their decisions influenced by the choices or advice of others
- *Policy-based choice*: choices can be made according to a specific policy (more common in an organizational setting)
- *Trial-and-error based choice*: a person may opt to randomly select an option to assess it (esp. when none of the above patterns leads to a clear decision)

User Simulation and Human Decision-making

- Cognitive Models
- Process Models
- Strategic Models
- Choice and Decision Making in Recommender Systems
- Mathematical Framework

Mathematical Framework

Markov decision process (MDP)

- Formally be described by a finite state space $\mathcal{S}$, a finite action set $\mathcal{A}$, a set of transition probabilities P, and a reward function R
- At a given point in time, the agent is in state $s \in \mathcal{S}$, and by executing action $a \in \mathcal{A}$, they transition into a new state s' according to the transition probability $P(s'|s,a)$ and receive reward $R(a,s)$
- The Markov property ensures that this transition depends only on the current state and action (which simplifies modeling and reduces computational complexity)

Example

Routing problems, such as the traveling salesman problem.

- Salesman = agent
- Routes available = the actions that the agent can take while in the current state
- Rewards = the costs of taking specific routes
- Goal = the optimal policy that lowers the overall cost for the entire duration of the trip

Using MDPs for User Simulation

- *State*: needs to encompass the high-level state in the information-seeking process, and the user's mental/cognitive state (goal, intent, preferences, emotional states, etc.)
- *Actions*: explicit and implicit actions the user might take
- *State transitions*: straightforward when we consider only explicit states and explicit actions
- *Reward (and Cost)*: models a user's objective of information seeking and the effort a user must make in order to achieve the goal
- *Policy*: determines how to choose an action in each state
    - Can be simple but interpretable models or machine-learned non-interpretable predictive models of user behaviour

Use of MDPs in RL vs. in User Simulation

Reinforcement learning

- The main focus revolves around finding an optimal policy (that maximizes the expected cumulative reward over time)
- Designing effective reward functions is crucial
- Transition probabilities are often observed from an external environment

User simulation

- Policy is based on an explicit model of user behaviour; does not need to be optimal, but needs to be controllable by the system designer
- The reward function can be used to encapsulate the costs and rewards based on observed data (from logs or user studies)
- Transition probabilities are also modeled explicitly based on some model of user behaviour

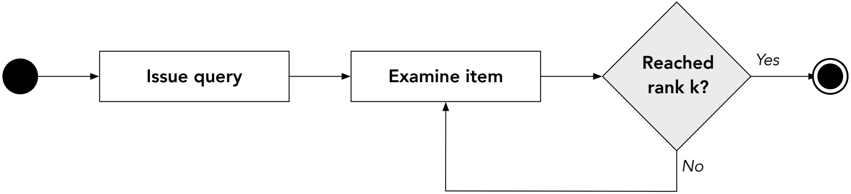
Simulating Interactions with Search and Recommender Systems

Simulating Interactions with Search and Recommender Systems

- Workflow Models
- Simulating Queries
- Simulating Scanning Behaviour
- Simulating Clicks
- Simulating Document Processing
- Simulating Stopping Behaviour
- Validating Simulators

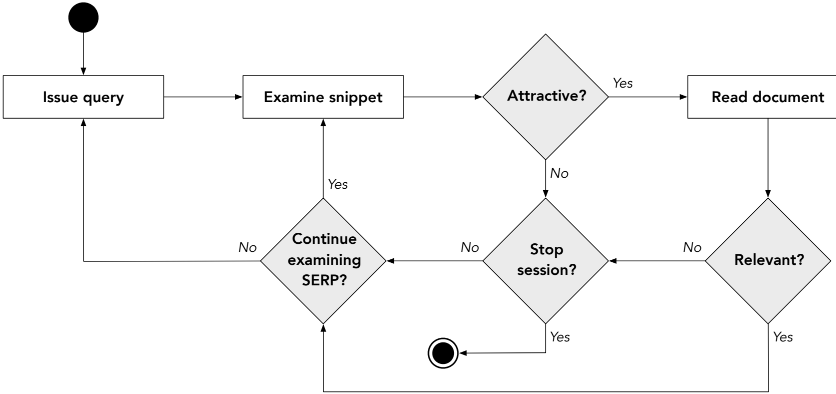
Workflow Models

- Simulation relies on simplified models (of workflows and user behaviour), which allows for "unnecessary complications" to be abstracted away
- The main research challenge is determining what elements of human behaviour to capture in these abstractions, while keeping the models as simple as possible



Naive searcher model, corresponding to highly abstracted user

Search Workflows

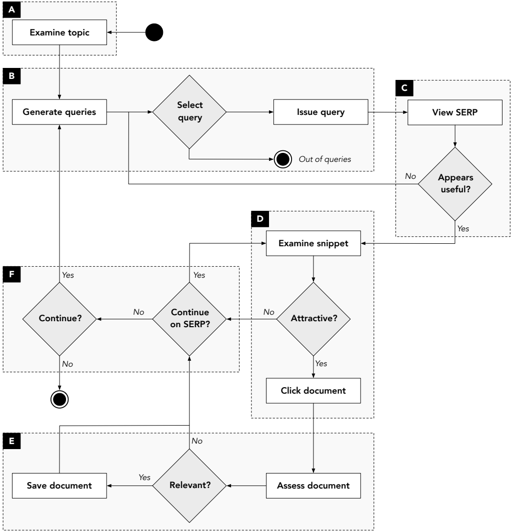


Searcher model by Baskaya et al. (2013)

Search Workflows

Complex Searcher Model, proposed by Maxwell et al. (2015) and then further updated in (Maxwell and Azzopardi, 2018)

(A) Topic examination
(B) Querying
(C) SERP examination
(D) Result summary examination
(E) Document examination
(F) Deciding to stop



Simulating Interactions with Search and Recommender Systems

- Workflow Models
- Simulating Queries
- Simulating Scanning Behaviour
- Simulating Clicks
- Simulating Document Processing
- Simulating Stopping Behaviour
- Validating Simulators

Simulating Queries

- Possible user goals”
  - To find some “known items” (*known item search*)
  - To find relevant information (*ad hoc search*)

Table: Overview of query generation approaches

Generation	**Reference**	**Input $\Rightarrow$ Output**	**Method**	
Individual queries	(Azzopardi et al., 2007)	$\emptyset \Rightarrow (q, d)$	Prob.	Stat.
	(Azzopardi, 2009)	$T = (q_0, R) \Rightarrow q$	Prob.	Stat.
Controlled query sets	(Jordan et al., 2006)	$R \Rightarrow \langle q_1, .., q_n \rangle$	Det.	Stat.
Query reformulations	(Baskaya et al., 2012)	$\{t_1, .., t_m\} \Rightarrow \langle q_1, ..q_n \rangle$	Det.	Stat.
	(Carterette et al., 2015)	$T = (s, Q)$, $S_{1..i-1} \Rightarrow q_i$	Prob.	Dyn.

Simulating Queries

Generating individual queries for **known item search** (Azzopardi et al., 2007)

- Initialize an empty query $q = \{\}$
- Sample a document d to be the known item with probability $P(d)$
- Select the query length l with probability $P(l)$
- Repeat s times:
  - Select a term t_i from the (unigram) language model of document d with probability $P(t_i|\theta_d)$
  - Add t_i to the query q
- Record (q, d) as the known-item query-document pair

Example Topic

<num> Number: 303
<title> Hubble Telescope Achievements

<desc> Description:
Identify positive accomplishments of the Hubble telescope since it was launched in 1991.

<narr> Narrative:
Documents are relevant that show the Hubble telescope has produced new data, better quality data than previously available, data that has increased human knowledge of the universe, or data that has led to disproving previously existing theories or hypotheses. Documents limited to the shortcomings of the telescope would be irrelevant. Details of repairs or modifications to the telescope without reference to positive achievements would not be relevant.

Example TREC topic definition (from Robust 2003 track). The terms present in such topic definitions are often used as the basis of query generation.

Simulating Queries

Generating query reformulations (Baskaya et al., 2012)

- It is assumed that a fixed set of terms $t_1, \ldots, t_m$ is available for each topic, from which queries may be constructed
- Five prototypical strategies, based on term level changes (grounded in observed real life behaviour)
  - **S1**: an initial single-term query is followed by queries that repeatedly replace that term: $q_1 = \{t_1\} \rightarrow q_2 = \{t_2\} \rightarrow q_3 = \{t_3\} \rightarrow \ldots$
  - **S2**: an initial two-term query is followed by queries repeatedly varying the second term: $q_1 = \{t_1, t_2\} \rightarrow q_2 = \{t_1, t_3\} \rightarrow q_3 = \{t_1, t_4\} \rightarrow \ldots$
  - **S3**: an initial three-term query is followed by queries repeatedly varying the third term: $q_1 = \{t_1, t_2, t_3\} \rightarrow q_2 = \{t_1, t_2, t_4\} \rightarrow q_3 = \{t_1, t_2, t_5\} \rightarrow \ldots$
  - **S4**: an initial single-term query is followed by queries which extend the previous query with a new term: $q_1 = \{t_1\} \rightarrow q_2 = \{t_1, t_2\} \rightarrow q_3 = \{t_1, t_2, t_3\} \rightarrow \ldots$
  - **S5**: an initial two-term query is followed by queries which extend the previous query with a new term: $q_1 = \{t_1, t_2\} \rightarrow q_2 = \{t_1, t_2, t_3\} \rightarrow q_3 = \{t_1, t_2, t_3, t_4\} \rightarrow \ldots$

Simulating Queries

Generating queries dynamically within search sessions (Carterette et al., 2015)

- It is assumed that topics $T = (s, Q)$ come with a textual description s and a set of queries Q (e.g., TREC Session track)
- Query length is conditioned on the topic
- language model from which query terms are sampled is continuously updated based on the results the user has seen for previous queries in the session

1. Generate n candidate queries:
    - Sample query length l according to $P(l|T)$
    - Iterate over terms in $P(t|T, l, i)$ in order of decreasing probability:
        - Flip a coin to decide whether to add t to the query
        - Repeat until l terms are sampled
2. Sample one query from the set according to $P(q|T)$ to be returned as the simulated query reformulation q_i

Simulating Interactions with Search and Recommender Systems

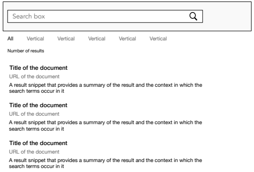
- Workflow Models
- Simulating Queries
- Simulating Scanning Behaviour
- Simulating Clicks
- Simulating Document Processing
- Simulating Stopping Behaviour
- Validating Simulators

Simulating Scanning Behaviour

- Concerned with how the user processes the list of results presented to them in response to their search query
- Commonly, **sequential browsing** is assumed
- *Cascade model* (Craswell et al., 2008)
  - The user examines each result and decides whether the snippet is deemed relevant enough to warrant a click
  - Snippets below a clicked result are not examined (i.e., the user would stop after having found a relevant result)
- *User browsing model* (Dupret and Piwowarski, 2008)
  - At each rank position, the user first decides whether to look at the snippet or not ("attractive" or not)
  - Then, resume the scan of the result list from the next rank position (whether the result gets clicked or not)
  - Models the event that user *examines* the snippet $(P(E = 1|R_i, C_1, ..., C_{i-1}))$ and, independently from it, whether they find the snippet *attractive* $(P(A = 1|R_i))$

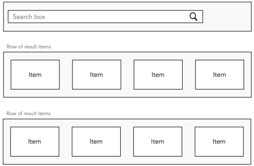
Complex Presentation Layouts

Current approaches rarely consider modern SERPs and alternative presentation layouts, where the top-down traversal assumption is challenged

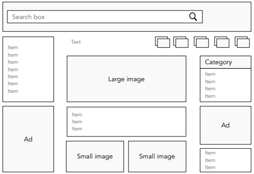


(a) A traditional “ten blue links” layout.

(b) A product search layout.



(c) A video recommendation layout.



(d) An advertisement layout.

Simulating Interactions with Search and Recommender Systems

- Workflow Models
- Simulating Queries
- Simulating Scanning Behaviour
- Simulating Clicks
- Simulating Document Processing
- Simulating Stopping Behaviour
- Validating Simulators

Simulating Clicks

- Mimic a user's decision on whether to click on a search result (to view it in detail) after being exposed to a result (snippet)
- Often integrated with the modeling of scanning behaviour
- Many tradeoffs to be made, especially interpretability vs. prediction accuracy
  - *Position-based simulation*: clicking probability only depends on the rank positions:
    - $P(Click = 1|Rank = i, R_1, R_2, ..., R_k) \approx P(Click = 1|Rank = i)$
    - Naive but generally applicable to any simulation scenario
  - *Content-based simulation*: snippet content is used to model the probability of clicking
    - Intuitively more accurate, but learned models are prone to overfitting and may lose interpretability
- *Perfect snippet assumption* (implicit): user is assumed to be able to tell whether a result is relevant based on the snippet and would always click on a result if it is relevant

Formal View of Click Modeling in MDP

- Click modeling $=$ modeling the policy of choosing between 0 (not clicking) and 1 (clicking) for the clicking action $A_C \in \{0, 1\}$
- Current state S_C includes all the relevant context information to this decision, including, e.g.,
  - a user's current query Q
  - the snippet R_i at the current position i
  - the whole ranked list of results, $R_1, R_2, ..., R_k$
  - any other useful information about the user U
  - any (historical) context information that might affect a user's decision on whether to click on a result H (e.g., historical interactions of the user U or other similar users)
- The clicking policy generates a value for A_C based on S_C: $A_C = \pi_C(S_C)$

An Overly Simplified Case

The policy uses only the current ranking position to determine whether to click a result. In this case,

- $\pi_C(S_C) \approx \pi_C(i)$, leading to a stochastic clicking policy specified based on a position-specific clicking probability
- Intuitively, a higher ranking position (i.e., a smaller i) would have a higher probability of clicking
- A clicking policy defined as $\pi_C(i) = 1/\log_2(i+1)$, would give us an interpretation of the discounting coefficients used in the nDCG evaluation measure as a naive clicking policy

Interaction of Click Modeling and Scanning

- With a separate model for scanning behavior, click modeling is based on the assumption that the user has already examined a snippet and would need to decide whether to click on it to further examine the content of the document
- The simulated clicking policy would only be used in simulating a user when the simulated scanning strategy has predicted examination of the result
- Scenarios of interaction of click modeling and examination of documents

Table: User interaction with search results: examination vs. clicking

Shown to user?	**Examined by user?**	**Clicked by user?**	**Status of result**
No	N/A	N/A	Unexposed result
Yes	No	N/A	Ignored result (affected by stopping strategy)
Yes	Yes	No	Skipped result (negative feedback)
Yes	Yes	Yes	Clicked result

Using Click Models in User Simulators

- Trade-off between click prediction accuracy and interpretability
    - More sophisticated models (e.g., based on deep learning), are more accurate in predicting clicks, but they are deficient in their interpretability $\Rightarrow$ hard to simulate variations of users
- Some models may not be realistic
    - Click decision is generally made based on the information shown in the result snippet of a result without having access to the whole document
    - User's prior background knowledge about the query topic is also relevant
        - For example, an expert user may be able to recognize a relevant document based on just a short snippet, where a novice user might not
- Specific click models
    - For search, see (Chuklin et al., 2015) for a review (e.g., based on the relevance level of the underlying document (Baskaya et al., 2013), using features of document titles, URLs, and snippets, which are available to users (Carterette et al., 2015), comparing the language model representing the user's background knowledge with a language model created from the snippet (Maxwell and Azzopardi, 2016))
    - Recommender systems: predicting user clicks for the purpose of optimizing recommendation accuracy, thus using as much context information as possible, including information not available to a real user

Simulating Interactions with Search and Recommender Systems

- Workflow Models
- Simulating Queries
- Simulating Scanning Behaviour
- Simulating Clicks
- Simulating Document Processing
- Simulating Stopping Behaviour
- Validating Simulators

Simulating Document Processing

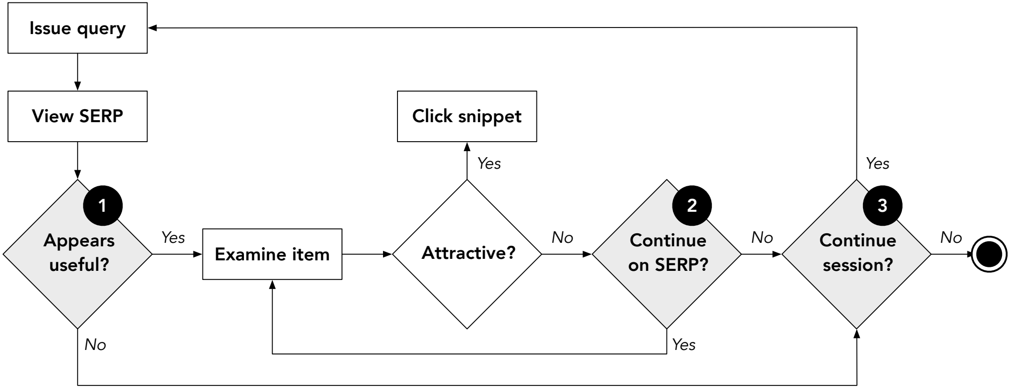
- Processing (i.e., reading and understanding) a document requires an **effort** from the user and yields some **utility** to them (enabling the user to acquire new information, thus changing cognitive state)
- *Dwell time* is often used as a proxy for effort
  - Time (in seconds) needed to process a document of length l, measured in words (Smucker and Clarke, 2012)
    $$T_D(l) = al + b$$
    User is reading at a rate of a seconds per word, and then uses a constant amount of b seconds to make an assessment about the document's relevance
- *Relevance* is used as a proxy for utility
  - Commonly, leveraging ground truth relevance assessments in existing test collections
  - Alternatively, predict whether the user would find the document relevant
    - Represent the user's knowledge state as a language model that evolves based on the documents encountered (Maxwell and Azzopardi, 2016)
  - Note that utility is meant to be a broader concept than topical relevance!
    - Includes quality, novelty, importance, credibility, etc.
    - Encompasses everything that the user values, e.g., a witty or engaging writing style

Simulating Interactions with Search and Recommender Systems

- Workflow Models
- Simulating Queries
- Simulating Scanning Behaviour
- Simulating Clicks
- Simulating Document Processing
- Simulating Stopping Behaviour
- Validating Simulators

Simulating Stopping Behaviour

Users can decide to stop the search process at various points



Excerpt from the updated Complex Searcher Model (Maxwell and Azzopardi, 2018), highlighting various stopping decision points: (1) SERP-level stopping, (2) query-level stopping, and (3) session-level stopping

Simulating Stopping Behaviour

- Several user studies (interviews) to understand *why* people decide to stop
- Users do not apply predetermined criteria, but rather base stopping decisions on the feeling of "good enough"
  - Factors include time constraints, diminishing returns of further information seeking, and increasing redundancy of information encountered
- Different heuristic rules to quantitatively characterize the sense of "good enough," for example,
  - *Satisfaction*: encountering a predefined number of relevant snippets
  - *Searcher frustration*: observing a certain number of non-relevant snippets
  - *Satisfaction or frustration*: stopping as soon as one of the two conditions is met
  - *Time-based*: total amount of time spent on the SERP or time elapsed after the last relevant document found

Simulating Interactions with Search and Recommender Systems

- Workflow Models
- Simulating Queries
- Simulating Scanning Behaviour
- Simulating Clicks
- Simulating Document Processing
- Simulating Stopping Behaviour
- Validating Simulators

Validating Simulators

- Validating whether the simulator imitates the behaviour of real users *sufficiently well*
- Would a simulated user lead to similar retrieval performance to what is obtained from real users?
  - E.g., simulated queries against real queries
- Would a simulated user produce data that matches the characteristics of real user data?
  - How well a user simulator can predict data observed in search logs (e.g., search session statistics)?
- Does the user simulator behave as expected for it intended use (e.g., for evaluating an interactive system)
  - Tester-based framework (Labhishetty and Zhai, 2021, 2022)
  - Tester: System A is expected to perform better than system B under a certain condition (e.g., for a certain kind of queries)
  - Simulator passes the test if the expected behavior is observed
  - Reliability of a user simulator and reliability of a Tester can be estimated jointly

Simulating Interactions with Conversational Assistants

Conversational AI

- High-level categorization of systems
  - *Goal-driven* (a.k.a. *task-oriented*): aiming to assist users to complete some specific task $\Leftarrow$ our focus
  - *Non-goal-driven* (a.k.a. *chatbots*): aiming to carry on an extended conversation ("chit-chat"), usually with the purpose on entertainment

Conversational AI

- High-level categorization of systems
  - *Goal-driven* (a.k.a. *task-oriented*): aiming to assist users to complete some specific task $\Leftarrow$ our focus
  - *Non-goal-driven* (a.k.a. *chatbots*): aiming to carry on an extended conversation ("chit-chat"), usually with the purpose on entertainment
- **Conversational information access: tasks with an underlying information need, which can be satisfied through a conversation**
  - Includes the tasks of search, recommendation, and question answering (boundaries often blurred)

Challenges

Traditional search and recommender systems	Conversational information access
Limited set of user actions allowed by the system's UI	User intents need to be inferred from free text
Interactions are either driven by the user (search) or by the system (recommendation)	*Mixed initiative*: the user and system both actively participate in addressing the user's information need
Results are restricted to a ranked list of items	Results can be text of arbitrary length (incl. semi-structured elements and questions posed to the user)

Challenges

Traditional search and recommender systems	Conversational information access
Limited set of user actions allowed by the system's UI	User intents need to be inferred from free text
Interactions are either driven by the user (search) or by the system (recommendation)	*Mixed initiative*: the user and system both actively participate in addressing the user's information need
Results are restricted to a ranked list of items	Results can be text of arbitrary length (incl. semi-structured elements and questions posed to the user)

$\Rightarrow$ More advanced natural language understanding capabilities are required

Conceptualization of Conversational Information Access

- Dialogue is a sequence of *turns*
- Each turn is a natural language *utterance* from either the user or the system
- *Dialogue act* represent the function or high-level intention of an utterance
  - Typically represented as tuples: *intent* and (optionally) slot-value pairs (e.g., `AFFIRM` or `INFORM(a=x,b=y,...)`)
  - The set of dialogue acts needs to be designed specific to the objectives of the dialogue application (various taxonomies exist)
- Taxonomy of user and system actions by Azzopardi et al. (2018)
  - Fn: conversational functionality according to (Radlinski and Craswell, 2017)
  - Pr: search process in (Trippas et al., 2018)

Fn.	Pr.	User actions	System actions	Fn.	
	Query formul.	**Reveal** - Disclose - Non-disclose - Revise - Refine - Expand	**Inquire** - Extract - Elicit - Clarify	User revealment	Memory
Set retrieval	Result exploration	**Inquire** - List - Summarize - Compare - Subset - Similar **Navigate** - Repeat - Back - More ... - Note	**Reveal** - List - Summarize - Compare - Subset - Similar **Traverse** - Repeat - Back - More ... - Record	System revealment	Memory
Mixed initiative		**Interrupt** - Interrupt **Interrogate** - Understand - Explain	**Suggest** - Recommend - Hypothesize **Explain** - Report - Reason	System revealment	Memory

Conceptualization: Dialogue Structure

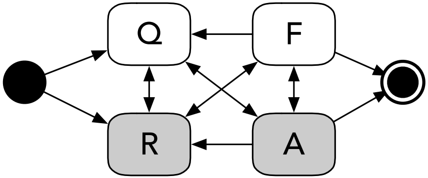
Dialogue structure: A characterization of dialogues in terms of overall organization, sequencing, and components.

- Three stages in e-commerce conversational search (Zhang et al., 2018)
  - Initiation, conversation, and display
- Mixed-initiative conversational search(Aliannejadi et al., 2021)
  - Querying, feedback, and browsing
- Transition patterns in information-seeking conversations (Qu et al., 2018)
  - START $\Rightarrow$ original question ($\Rightarrow$ potential answer $\Rightarrow$ further details)x3 $\Rightarrow$ potential answer $\Rightarrow$ positive feedback $\Rightarrow$ END
- Context-driven recommendation in the restaurant domain (Lyu et al., 2021)
  - (1) Preference elicitation and refinement in the first stage, (2) inquiry and critiquing in subsequent stages, (3) additional comparisons

Conceptualization

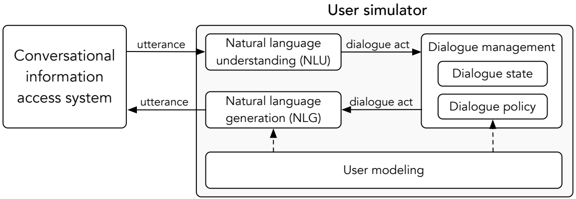
QRFA (Vakulenko et al., 2019): generic model of conversational information seeking processes.

- Four basic classes: 2 for user and 2 for system (proactive and reactive)
  - User: Query and Feedback
  - System: Request and Answer



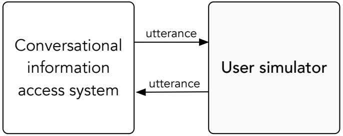
Simulator Architectures

Modular systems



- Model user responses semantically on the level of dialogue acts, then generate the corresponding natural language utterances

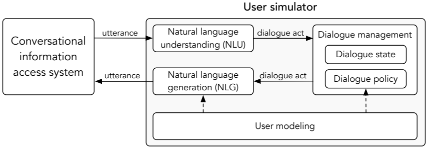
End-to-end systems



- Operate on the utterance level (generate textual responses directly)
- Might yield more fluent dialogues, but do not allow for interpretable user behaviour

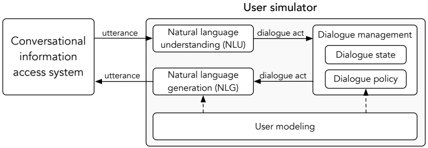
Modular Systems

- *Natural language understanding (NLU)*: converting the (raw) system utterance into an internal semantic representation (dialogue act)
  - *Intent detection* is naturally approached as a classification task
  - *Slot filling* is a sequence labelling problem
- *Dialogue management*: maintaining the dialogue state and determining the next user action
  - The *dialogue state* is based on the notion of a *semantic frame*: collection of slots that together specify what the system needs to know to complete a given task
  - The *dialogue policy* determines how the user should respond



Modular Systems

- *Natural language generation (NLG)*: turning the generated response from a structured representation (dialogue act) into natural language
  - Template-based, retrieval-based, text generation, and hybrid methods
- *User modeling*: capturing the characteristics of individuals that would influence how they interact with the system
  - Information about the user's goal, knowledge, preferences, personal characteristics (e.g., patience), and beliefs about the system



User Dialogue Policy

- Here: task-oriented dialogue in a restricted "slot-filling" sense
  - A *domain ontology* describes the specific intents, slots, and entities that can be talked about
  - The user can specifying their constraints in terms of *informable slots* and requesting information on *requestable slots*
  - Appropriate for modeling user goals in some scenarios (e.g., item recommendation), while others (e.g., exploratory search) are open research problems
- Dialogue is represented as a sequence dialogue acts by the system (a_i^s) and the user (a_i^u) as they take turns: $a_0^s \rightarrow a_0^u \rightarrow a_1^s \rightarrow a_1^u \rightarrow \cdots \rightarrow a_{t-1}^s \rightarrow a_t^s$
- The policy π determines what action a_{t+1}^u the user should take next, given the dialogue history

Statistical User Models: N-grams Models (Eckert et al., 1997)

- Next response based on the dialogue history (resembling the estimation of language models):

$$\pi(s_t) = P(a^u_{t+1} | a^s_t, a^u_t, a^s_{t-1}, a^u_{t-1}, \ldots, a^u_0, a^s_0)$$

- Strong simplifying assumption to condition the next user action exclusively on the preceding system action:

$$\pi(s_t) = P(a^u_{t+1} | a^s_t)$$

- Conditional probabilities estimated from an annotated dialogue corpus
- No information about the user's goal, no constraints on the simulated user behaviour $\Rightarrow$ fails to produce realistic dialogues
  - Placing constraints on the dialogue flow yields somewhat more realistic dialogues (Levin et al., 2000), but the consistency between user responses across the dialogue is still not guaranteed

Statistical User Models: Goal-directed User Model with Memory (Pietquin, 2004)

- Explicit representation of the user goal as a sequence of slot-value pairs with priority: $G = \langle (slot_1, value_1, prior_1), \ldots, (slot_n, value_n, prior_n) \rangle$
  - When the user is prompted for the relaxation of some attribute, slot-value pairs with a higher priority are less likely to be relaxed
- Dialogue history at time t is represented as a vector $h_t = \langle c_1, \ldots, c_n \rangle$
  - c_i is the count of the occurrences a value is provided for the corresponding $slot_i$
  - Enables the simulator to disclose new information to the system if mixed initiative is supported
- Allows for automatic evaluation in terms of full or partial task completion (given how goals are represented)

Statistical User Models: Agenda-based Simulator (Schatzmann et al., 2007)

- Factors the user state into an agenda and a goal $s_t = (A_t, G_t)$
- Agenda A_t is a stack-like structure, representing the pending intentions of the user
- Goal is a tuple $G_t = (C_t, R_t)$, where
  - C_t is a set of domain-specific constraints the user wants to impose on the dialogue
  - R_t specify requests, i.e., slots whose values are initially unknown to the user and will need to be filled out during the conversation
- For example (restaurant recommendation): looking for the name, address, and phone number of a centrally located bar serving beer:

$$C_0 = \begin{bmatrix} \text{type} & = & \text{bar} \\ \text{drinks} & = & \text{beer} \\ \text{area} & = & \text{central} \end{bmatrix} \qquad R_0 = \begin{bmatrix} \text{name} & = & \\ \text{addr} & = & \\ \text{phone} & = & \end{bmatrix}$$

Statistical User Models: Agenda-based Simulator (Schatzmann et al., 2007)

- Agenda initialization
  - All goal constraints set to `INFORM` acts and all goal requests set to `REQUEST` acts
  - `BYE` added at the bottom of the agenda to close the dialogue

$$A_0 = \begin{bmatrix} \text{INFORM}(\text{type} = \text{bar}) \\ \text{INFORM}(\text{drinks} = \text{beer}) \\ \text{INFORM}(\text{area} = \text{central}) \\ \text{REQUEST}(\text{name}) \\ \text{REQUEST}(\text{addr}) \\ \text{REQUEST}(\text{phone}) \\ \text{BYE} \end{bmatrix}$$

- As the conversation progresses, the agenda and goal are dynamically updated
  - Next user action simplifies to popping items from the top of the agenda
  - Agenda updates are push operations, where dialogue acts get added on top of the agenda

Sequence-to-sequence Models

More recently, learning user simulators fully data-driven from dialogue corpora

Reference	Architecture	Input	Output	Modeling goal?	Multi-domain?
(El Asri et al., 2016)	RNN-LSTM	feature vect.	dialogue act	Y	N
(Gür et al., 2018)	RNN-GRU	dialogue act	dialogue act	Y	N
(Lin et al., 2021)	Transformer	feature vect.	dialogue act	Y	Y
(Crook and Marin, 2017)	RNN-GRU/LSTM	utterance	utterance	N	N
(Kreyssig et al., 2018)	RNN-LSTM	feature vect.	utterance	Y	N
(Lin et al., 2022)	Transformer	context	dial. act + utt.	Y	Y

- Operating on the semantic level of dialogue acts vs. text utterances directly
- From manual feature engineering to progressively adopting end-to-end approaches
    - Interpretability diminishes, limited control over the behaviour of the simulated user
    - Effectively, only indirect control through the input training data provided

Sequence-to-sequence Models

Representation of conversation contexts (i.e., dialogue state).

- (El Asri et al., 2016): at turn t, simulator takes $\langle c_1, \ldots, c_t \rangle$ as input, where c_t consists of four components (all represented as binary vectors)
  - Most recent machine action
  - Inconsistency between machine information and user goal (i.e., slots that have been misunderstood by the system so that these may be corrected)
  - Constraint status (to inform the system about preferences)
  - Request status (to keep track of requests that have not yet been fulfilled)
- (Gür et al., 2018): encode the entire dialogue history based on the user goal and system dialogue act. System dialogue acts are represented on a more coarse level by replacing specific slot values with one of the following:
  - `Requested`, if the value is requested by the system
  - `ValueInGoal`, if the value appears in the user goal
  - `ValueContradictsGoal`, if the value contradicts the user goal
  - `DontCare`, if the value in the user goal is flexible
  - `Other` otherwise

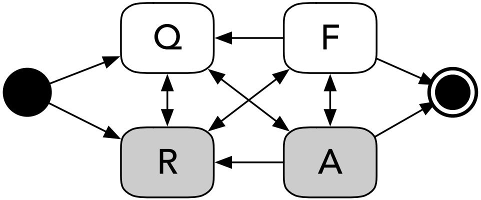
User Simulation for Conversational Information Access

- Conversational information access is a broad task that encompasses the goals of conversational search, recommendation, and question answering
- Approaches that support this holistic view are yet to be developed
  - Appropriate datasets have only been recently started to become available (Bernard and Balog, 2023)
- As of now, there are no multi-goal simulators, simulators are developed in a goal-specific manner:
  - Conversational search
  - Conversational recommendation

User Simulation for Conversational Search

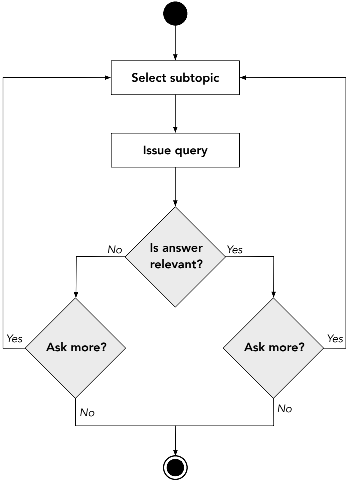
Two main types of user utterances considered:

- User-initiated questions (Query)
- Responses to system-initiated questions (Feedback)



Simulating User Questions (Lipani et al., 2021)

- It is assumed that the user's goal is to learn about a set of subtopics by interacting with the system
- Both user queries and system responses are represented as *subtopics*
- At each dialogue turn the user asks about a particular subtopic
- Based on the relevance of the system's response, the user will ask further questions (about the same subtopic or a different one) or stop querying



Simulating User Questions (Lipani et al., 2021)

- The user dialogue policy is based on the notion of persistence in querying the system, depending on the relevance of the answer to the previous query
- Start with a query in turn 1
- For any subsequent turn t
  - Leave with probability $P(L_t = l | Q_t = q, R_t = r)$ if the system response was relevant
  - Leave with probability $P(L_t = l | Q_t = q, R_t = \bar{r})$ if the result was not relevant
  - Both probabilities are estimated from user logs
- Overall, the following data components are required:
  - A sample of information needs (i.e., topics)
  - For each topic, a pre-defined set of subtopics
  - Subtopic-level relevance judgments
  - A dialogue dataset with subtopic annotations for the estimation of state transition probabilities

Simulating Answers to Clarifying Questions (Salle et al., 2021)

- Simulating how a user would respond to clarifying questions that are in the form: "Are you looking for *[facet]*?"
- *User intent model*: represents the user's information need and estimates whether the clarifying question matches the user's intent
  - Implemented by fine-tuning a BERT model for binary classification
- *Persona model*: specifies personal user characteristics
  - *Cooperativeness* ($\in [0, 1]$): the user's willingness to help the system by giving an informative answer (e.g., "No, I'm looking for *[intent]*") vs. simply "Yes" or "No")
  - *Patience*: maximum effort (number of turns) the user is willing to spend interacting with the system

Simulating Answers to Clarifying Questions (Sekulić et al., 2022)

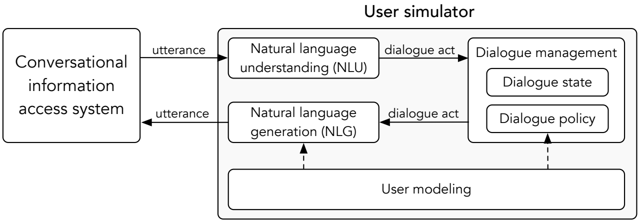
- Fine-tuning a transformer-based large language model (LLM) for the task of answering clarifying questions
- DoubleHead GPT-2 with language modeling and classification losses
- Training input part 1 is given as the sequence `in[SEP]q[SEP]cq[bos]a[eos]`
  - in: textual description of the user's information need
  - q: user's query
  - cq: clarifying question asked by the system
  - a: answer given by the user
  - `[bos]` and `[eos]` are special tokens indicating the beginning and end of a sequence
  - `[SEP]` is a separation token
- Training input part 2: distractor answer and a binary label indicating which of the answers is preferable
  - Distractor answers are sampled from the training dataset heuristically
  - E.g., if the answer starts with "Yes" then the distractor answer starts with "No"
- At inference time, the above input sequence is given without the answer segment, which will be generated by the LLM

User Simulation for Conversational Search (Owoicho et al., 2023)

- Generating a variety of utterances by few-shot prompting a ChatGPT model:
  - Queries to seek information
  - Answers to clarifying questions
  - Feedback to system responses
- Note: LLM-based approaches generate answers that are fluent and natural-sounding, they work much like black boxes
  - The behaviour of the simulated user can be controlled only indirectly and only to a certain extent via training examples

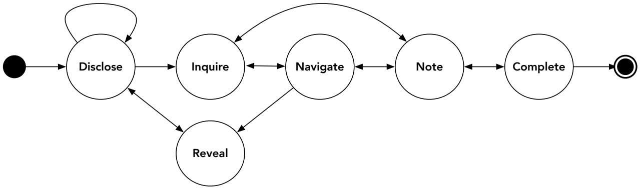
User Simulation for Conversational Recommendation (Zhang and Balog, 2020)

- Task: elicit user preferences using natural language interactions, point users to potential items of interest, and process feedback by users on the made suggestions
- Can naturally be framed in the classical sense of task-oriented dialogue systems:
  - Find items that satisfy the set of constraints expressed by the user, which can be represented in terms of slot-value pairs: $C = \langle (slot_1, value_1), \ldots, (slot_n, value_n) \rangle$
- Use a modular simulator architecture



User Simulation for Conversational Recommendation (Zhang and Balog, 2020)

- *NLU*: utilize the fact that many conversational systems use a limited set of language expressions (often as a result of a template-based NLG)
  - A small sample of annotated dialogues from a given system is sufficient
- *Dialogue policy*: agenda-based, guided by an *interaction model*
  - Interaction model specifies the set of user actions and expected system response for each user action
  - The latter allows the simulator to determine whether the system responds to the user with an appropriate action (i.e., "understood" the user)



User Simulation for Conversational Recommendation (Zhang and Balog, 2020)

- *User model*: based on a *preference model*, which is a a knowledge structure with $(slot, value, pref)$ triples
  - Grounded in actual user preferences, by randomly sampling a user, then subsampling item ratings of that user from a dataset of historical user-item interactions
  - The rest of the ratings are used as held-out data for automatic evaluation
  - To ensure the consistency of preferences, a *personal knowledge graph* is used
- *NLG*: based on templates, using a number of different articulations for each intent

Validation

- *Individual utterances*: commonly, human raters evaluate the generated responses along different dimensions (e.g., naturalness, usefulness, grammar)
- *Individual dialogues*: side-by-side human evaluation protocol (Zhang and Balog, 2020)
  - Assessors are given transcripts of two conversations, in random order
  - They have to guess which of the two is the generated by a human
- *A collection of generated dialogues*:
  - *High-level dialogue features*: avg. dialogue length, ratio of user vs. system actions, etc.
  - *Dialogue style*: distribution of dialogue acts, user cooperativeness (proportion of slot values provided when requested), etc.
  - *Dialogue efficiency*: success (or task completion) rate, reward, completion time, etc.

Validation

Ultimately: how well can simulation predict the performance of a system with real users?

Method	Reward	Success Rate
Real users	**A** (8.88) > **B** (7.56) > **C** (6.04)	**B** (0.864) > **A** (0.833) > **C** (0.727)
QRFA-Single	**A** (8.04) > **B** (7.41) > **C** (6.30)	**B** (0.836) > **A** (0.774) > **C** (0.718)
CIR6-Single	**A** (8.64) > **B** (8.28) > **C** (6.01)	**B** (0.822) > **A** (0.807) > **C** (0.712)
CIR6-PKG	**A** (11.12) > **B** (10.65) > **C** (9.31)	**A** (0.870) > **B** (0.847) > **C** (0.784)

Performance of conversational agents using real vs. simulated users in (Zhang and Balog, 2020)

Conclusion and Future Challenges

Summary

- There is a critical need for sound and scalable means of automatic evaluation of information access systems
- Benefits of using user simulation for system evaluation
  - Enables reproducible experiments with evaluation of interactive information access systems
  - Allows to test their systems under various scenarios and conditions, which may be difficult or impossible to achieve in real-world testing
  - Can help identify potential flaws or weaknesses in a system before it is deployed
- Most work on user simulation has been done for search engines, less so for recommender systems, but increasingly more common for conversational assistants
- Lot of component-level solutions; integrating these into a coherent and holistic user simulator remains a future challenge

Future Direction: Embracing Simulation-based Evaluation

- Simulation-based evaluation has not been widely adopted in the IR and RecSys communities
- Could be due to several factors:
  - Complexity of creating realistic simulations
  - Lack of consensus on simulation-based evaluation methodology
  - Open questions regarding the validity of simulations
  - Resources required to develop and run simulations
- Next steps
  - Leverage existing test collections and turn them into user simulators
  - Organize evaluation activities regularly (e.g., at TREC) for evaluating both user simulators and using simulation to evaluate IR systems

Future Direction: Fostering Industry-academia Collaboration

- User simulation is a technology that can help to foster collaboration between academia and industry
- Academia: Access to realistic datasets for evaluation is always a major challenge
- Industry: It is difficult to release datasets (e.g., due to privacy concerns)
- Releasing user simulators trained/estimated using commercial search log data should have much less privacy concerns than releasing any log data (directly)
- Self-sustainable innovation ecosystem
  - Academic researchers develop models/algorithms for user simulation and make them available as open source
  - Commercial service providers train and validate user simulators against their logs, and publish the trained simulators (without having to share any actual user data)
  - Academic researchers can develop and validate new search and recommendation algorithms against published simulators
  - Service providers get access to the most advanced algorithms developed by (external) researchers

Key Technical Challenge: Realisticity

- Informally, it is easy to understand what it means to simulate a user computationally
- Mathematically defining the problem remains a major open challenge (e.g., behavior similarity vs. model similarity)

"It remains an open question as to how realistic (i.e. human-like) simulators can be, or indeed should be. It is important to note that simulators do not need to be perfect mirrors of human behaviour, but instead simply need to be "good enough." By this, we mean that output from simulations should correlate well with human assessments on a given task with respect to some evaluation metric. The main requirement is reproducibility." – Sim4IR workshop (Balog et al., 2022)

Opportunities for Interdisciplinary Research

User Simulation overlaps with multiple related areas

- Information Retrieval: Conversational Search
- Recommender Systems: Conversational recommendation
- Agent Systems: Conversational task assistants
- Machine Learning: Reinforcement Learning
- HCI and Psychology: Simulators as Testable Hypotheses about Users
- Natural Language Processing: User Simulation and Large Language Models

Discussion

References

Mohammad Aliannejadi, Leif Azzopardi, Hamed Zamani, Evangelos Kanoulas, Paul Thomas, and Nick Craswell. 2021. Analysing Mixed Initiatives and Search Strategies during Conversational Search. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21)*. 16–26. https://doi.org/10.1145/3459637.3482231

Leif Azzopardi. 2009. Query Side Evaluation: An Empirical Analysis of Effectiveness and Effort. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*. 556–563. https://doi.org/10.1145/1571941.1572037

Leif Azzopardi and Maarten de Rijke. 2006. Automatic Construction of Known-Item Finding Test Beds. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*. 603–604. https://doi.org/10.1145/1148170.1148276

Leif Azzopardi, Maarten de Rijke, and Krisztian Balog. 2007. Building Simulated Queries for Known-Item Topics: An Analysis Using Six European Languages. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. 455–462. https://doi.org/10.1145/1277741.1277820

Leif Azzopardi, Mateusz Dubiel, Martin Halvey, and Jeffery Dalton. 2018. Conceptualizing Agent-Human Interactions During the Conversational Search Process. In *Proceedings of the 2nd International Workshop on Conversational Approaches to Information Retrieval (CAIR '18)*.

Krisztian Balog, David Maxwell, Paul Thomas, and Shuo Zhang. 2022. Report on the 1st Simulation for Information Retrieval Workshop (Sim4IR 2021) at SIGIR 2021. *SIGIR Forum* 55, 2, Article 10 (mar 2022). https://doi.org/10.1145/3527546.3527559

References

Feza Baskaya, Heikki Keskustalo, and Kalervo Järvelin. 2012. Time Drives Interaction: Simulating Sessions in Diverse Searching Environments. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12)*. 105–114. https://doi.org/10.1145/2348283.2348301

Feza Baskaya, Heikki Keskustalo, and Kalervo Järvelin. 2013. Modeling Behavioral Factors in Interactive Information Retrieval. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM '13)*. 2297–2302. https://doi.org/10.1145/2505515.2505660

Marcia J. Bates. 1989. The Design of Browsing and Berrypicking Techniques for the Online Search Interface. *Online Review* 13 (1989), 407–424. Issue 5. https://doi.org/10.1108/eb024320

Nicholas J Belkin, Robert N Oddy, and Helen M Brooks. 1982. ASK for Information Retrieval: Part I. Background and Theory. *Journal of Documentation* (1982).

Nolwenn Bernard and Krisztian Balog. 2023. MG-ShopDial: A Multi-Goal Conversational Dataset for e-Commerce. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*. https://doi.org/10.1145/3539618.3591883

Christine L. Borgman. 1996. Why are Online Catalogs Still Hard to Use? *Journal of the American Society for Information Science* 47, 7 (1996), 493–503. https://doi.org/10.1002/(SICI)1097-4571(199607)47:7<493::AID-ASI3>3.0.CO;2-P

Ben Carterette, Ashraf Bah, and Mustafa Zengin. 2015. Dynamic Test Collections for Retrieval Evaluation. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval (ICTIR '15)*. 91–100. https://doi.org/10.1145/2808194.2809470

References

Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. 2015. *Click Models for Web Search*. Morgan & Claypool. https://doi.org/10.2200/S00654ED1V01Y201507ICR043

C. Cleverdon and M. Kean. 1968. Factors Determining the Performance of Indexing Systems. Aslib Cranfield Research Project, Cranfield, England.

Michael D. Cooper. 1973. A Simulation Model of an Information Retrieval System. *Information Storage and Retrieval* 9, 1 (1973), 13–32. https://doi.org/10.1016/0020-0271(73)90004-1

Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An Experimental Comparison of Click Position-Bias Models. In *Proceedings of the 2008 International Conference on Web Search and Data Mining (WSDM '08)*. 87–94. https://doi.org/10.1145/1341531.1341545

Paul Crook and Alex Marin. 2017. Sequence to Sequence Modeling for User Simulation in Dialog Systems. In *Proceedings of Interspeech 2017*. 1706–1710. https://doi.org/10.21437/Interspeech.2017-161

Georges E. Dupret and Benjamin Piwowarski. 2008. A User Browsing Model to Predict Search Engine Click Data from Past Observations. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08)*. 331–338. https://doi.org/10.1145/1390334.1390392

W. Eckert, E. Levin, and R. Pieraccini. 1997. User Modeling for Spoken Dialogue System Evaluation. In *1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings*. 80–87. https://doi.org/10.1109/ASRU.1997.658991

References

Layla El Asri, Jing He, and Kaheer Suleman. 2016. A Sequence-to-Sequence Model for User Simulation in Spoken Dialogue Systems. In *Proceedings of Interspeech 2016*. 1151–1155. https://doi.org/10.21437/Interspeech.2016-1175

Izzeddin Gür, Dilek Hakkani-Tür, Gokhan Tür, and Pararth Shah. 2018. User Modeling for Task Oriented Dialogues. In *2018 IEEE Spoken Language Technology Workshop (SLT '18)*. 900–906. https://doi.org/10.1109/SLT.2018.8639652

Donna Harman. 1992. Relevance Feedback Revisited. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '92)*. 1–10. https://doi.org/10.1145/133160.133167

Peter Ingwersen and Kalervo Järvelin. 2005. *The Turn: Integration of Information Seeking and Retrieval in Context*. The Information Retrieval Series, Vol. 18. Springer. https://doi.org/10.1007/1-4020-3851-8

Anthony Jameson, Bettina Berendt, Silvia Gabrielli, Federica Cena, Cristina Gena, Fabiana Vernero, and Katharina Reinecke. 2014. Choice Architecture for Human-Computer Interaction. *Foundations and Trends in Human-Computer Interaction* 7, 1–2 (oct 2014), 1–235. https://doi.org/10.1561/1100000028

Kalervo Järvelin, Susan L. Price, Lois M. L. Delcambre, and Marianne Lykke Nielsen. 2008. Discounted Cumulated Gain Based Evaluation of Multiple-Query IR Sessions. In *Proceedings of the 30th European Conference on Advances in Information Retrieval (ECIR '08)*. 4–15.

Chris Jordan, Carolyn Watters, and Qigang Gao. 2006. Using Controlled Query Generation to Evaluate Blind Relevance Feedback Algorithms. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL '06)*. 286–295. https://doi.org/10.1145/1141753.1141818

References

Evangelos Kanoulas, Ben Carterette, Paul D. Clough, and Mark Sanderson. 2011. Evaluating Multi-Query Sessions. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*. 1053–1062. https://doi.org/10.1145/2009916.2010056

Heikki Keskustalo, Kalervo Järvelin, and Ari Pirkola. 2008. Evaluating the Effectiveness of Relevance Feedback Based on a User Simulation Model: Effects of a User Scenario on Cumulated Gain Value. *Information Retrieval* 11, 3 (June 2008), 209–228. https://doi.org/10.1007/s10791-007-9043-7

Florian Kreyssig, Iñigo Casanueva, Paweł Budzianowski, and Milica Gašić. 2018. Neural User Simulation for Corpus-based Policy Optimisation of Spoken Dialogue Systems. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL '18)*. 60–69. https://doi.org/10.18653/v1/W18-5007

Carol C. Kuhlthau. 1991. Inside the Search Process: Information Seeking from the User's Perspective. *Journal of the American Society for Information Science* 42, 5 (1991), 361–371. https://doi.org/10.1002/(SICI)1097-4571(199106)42:5<361::AID-ASI6>3.0.CO;2-%23

Sahiti Labhishetty and Chengxiang Zhai. 2021. An Exploration of Tester-Based Evaluation of User Simulators for Comparing Interactive Retrieval Systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21)*. 1598–1602. https://doi.org/10.1145/3404835.3463091

Sahiti Labhishetty and ChengXiang Zhai. 2022. RATE: A Reliability-Aware Tester-Based Evaluation Framework of User Simulators. In *Proceedings of the 44th European Conference on IR Research (ECIR '22)*. 336–350. https://doi.org/10.1007/978-3-030-99736-6_23

References

Anton Leuski. 2000. Relevance and Reinforcement in Interactive Browsing. In *Proceedings of the Ninth International Conference on Information and Knowledge Management (CIKM '00)*. 119–126. https://doi.org/10.1145/354756.354809

Esther Levin, Roberto Pieraccini, and Wieland Eckert. 2000. A Stochastic Model of Human-machine Interaction for Learning Dialog Strategies. *IEEE Trans. Speech Audio Process.* 8, 1 (2000), 11–23. https://doi.org/10.1109/89.817450

Hsien-chin Lin, Christian Geishauser, Shutong Feng, Nurul Lubis, Carel van Niekerk, Michael Heck, and Milica Gasic. 2022. GenTUS: Simulating User Behaviour and Language in Task-oriented Dialogues with Generative Transformers. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL '22)*. 270–282.

Hsien-chin Lin, Nurul Lubis, Songbo Hu, Carel van Niekerk, Christian Geishauser, Michael Heck, Shutong Feng, and Milica Gasic. 2021. Domain-independent User Simulation with Transformers for Task-oriented Dialogue Systems. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL '21)*. 445–456.

Aldo Lipani, Ben Carterette, and Emine Yilmaz. 2021. How Am I Doing?: Evaluating Conversational Search Systems Offline. *ACM Trans. Inf. Syst.* 39, 4, Article 51 (Aug. 2021), 22 pages. https://doi.org/10.1145/3451160

Shengnan Lyu, Arpit Rana, Scott Sanner, and Mohamed Reda Bouadjenek. 2021. A Workflow Analysis of Context-Driven Conversational Recommendation. In *Proceedings of the Web Conference 2021 (WWW '21)*. 866–877. https://doi.org/10.1145/3442381.3450123

References

Gary Marchionini. 1995. *Information Seeking in Electronic Environments*. Cambridge University Press. https://doi.org/10.1017/CBO9780511626388

David Maxwell and Leif Azzopardi. 2016. Agents, Simulated Users and Humans: An Analysis of Performance and Behaviour. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. 731–740. https://doi.org/10.1145/2983323.2983805

David Maxwell and Leif Azzopardi. 2018. Information Scent, Searching and Stopping: Modelling SERP Level Stopping Behaviour. In *Proceedings of the 40th European Conference on IR Research (ECIR '18)*. 210–222. https://doi.org/10.1007/978-3-319-76941-7_16

David Maxwell, Leif Azzopardi, Kalervo Järvelin, and Heikki Keskustalo. 2015. Searching and Stopping: An Analysis of Stopping Rules and Strategies. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM '15)*. 313–322. https://doi.org/10.1145/2806416.2806476

Vicki L. O'Day and Robin Jeffries. 1993. Orienteering in an Information Landscape: How Information Seekers Get from Here to There. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93)*. 438–445. https://doi.org/10.1145/169059.169365

Paul Owoicho, Ivan Sekulic, Mohammad Alianejadi, Jeffery Dalton, and Fabio Crestani. 2023. Exploiting Simulated User Feedback for Conversational Search: Ranking, Rewriting, and Beyond. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*. https://doi.org/10.1145/3539618.3591683

References

Olivier Pietquin. 2004. *A Framework for Unsupervised Learning of Dialogue Strategies*. Ph. D. Dissertation. Faculté Polytechnique de Mons, Belgium.

Peter Pirolli and Stuart Card. 1999. Information Foraging. *Psychological Review* 106 (1999), 643–675. Issue 4. https://doi.org/10.1037/0033-295X.106.4.643

Chen Qu, Liu Yang, W. Bruce Croft, Johanne R. Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and Characterizing User Intent in Information-Seeking Conversations. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '18)*. 989–992. https://doi.org/10.1145/3209978.3210124

Filip Radlinski and Nick Craswell. 2017. A Theoretical Framework for Conversational Search. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval (CHIIR '17)*. 117–126. https://doi.org/10.1145/3020165.3020183

Alexandre Salle, Shervin Malmasi, Oleg Rokhlenko, and Eugene Agichtein. 2021. Studying the Effectiveness of Conversational Search Refinement Through User Simulation. In *Proceedings of the 43rd European Conference on IR Research (ECIR '21)*. 587–602. https://doi.org/10.1007/978-3-030-72113-8_39

G. Salton. 1970. Evaluation problems in Interactive Information Retrieval. *Information Storage and Retrieval* 6, 1 (1970), 29–44. https://doi.org/10.1016/0020-0271(70)90011-2

Jost Schatzmann, Blaise Thomson, Karl Weilhammer, Hui Ye, and Steve Young. 2007. Agenda-Based User Simulation for Bootstrapping a POMDP Dialogue System. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers (NAACL-HLT '07)*. 149–152.

References

Jost Schatzmann, Karl Weilhammer, Matt Stuttle, and Steve Young. 2006. A Survey of Statistical User Simulation Techniques for Reinforcement-Learning of Dialogue Management Strategies. *The Knowledge Engineering Review* 21, 2 (June 2006), 97–126.

Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. 2022. Evaluating Mixed-Initiative Conversational Search Systems via User Simulation. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining (WSDM '22)*. 888–896. https://doi.org/10.1145/3488560.3498440

Catherine L. Smith and Paul B. Kantor. 2008. User Adaptation: Good Results from Poor Systems. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08)*. 147–154. https://doi.org/10.1145/1390334.1390362

Mark D. Smucker and Charles L.A. Clarke. 2012. Time-Based Calibration of Effectiveness Measures. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12)*. 95–104. https://doi.org/10.1145/2348283.2348300

Karen Spärck Jones. 1979. Search Term Relevance Weighting given Little Relevance Information. *Journal of Documentation* 35, 1 (1979), 30–48. https://doi.org/10.1108/eb026672

Louise T. Su. 1992. Evaluation measures for interactive information retrieval. *Information Processing & Management* 28, 4 (1992), 503–516. https://doi.org/10.1016/0306-4573(92)90007-M

Johanne R. Trippas, Damiano Spina, Lawrence Cavedon, Hideo Joho, and Mark Sanderson. 2018. Informing the Design of Spoken Conversational Search: Perspective Paper. In *Proceedings of the 2018 Conference on Human Information Interaction and Retrieval (CHIIR '18)*. 32–41. https://doi.org/10.1145/3176349.3176387

References

Andrew Turpin and Falk Scholer. 2006. User Performance versus Precision Measures for Simple Search Tasks. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*. 11–18. https://doi.org/10.1145/1148170.1148176

Andrew Turpin, Falk Scholer, Kalvero Jarvelin, Mingfang Wu, and J. Shane Culpepper. 2009. Including Summaries in System Evaluation. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*. 508–515. https://doi.org/10.1145/1571941.1572029

Svitlana Vakulenko, Kate Revoredo, Claudio Di Ciccio, and Maarten de Rijke. 2019. QRFA: A Data-Driven Model of Information-Seeking Dialogues. In *Proceedings of the 41st European Conference on IR Research (ECIR '19)*. 541–557. https://doi.org/10.1007/978-3-030-15712-8_35

Yiming Yang and Abhimanyu Lad. 2009. Modeling Expected Utility of Multi-Session Information Distillation. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory (ICTIR '09)*. 164–175. https://doi.org/10.1007/978-3-642-04417-5_15

Steve Young. 1999. Probabilistic Methods in Spoken Dialogue Systems. *Philosophical Transactions of the Royal Society (Series A)* 358 (1999), 1389–1402. Issue 1769.

Steve Young, Milica Gašić, Simon Keizer, FranÁois Mairesse, Jost Schatzmann, Blaise Thomson, and Kai Yu. 2010. The Hidden Information State Model: A Practical Framework for POMDP-based Spoken Dialogue Management. *Computer Speech & Language* 24, 2 (2010), 150–174. https://doi.org/10.1016/j.csl.2009.04.001

References

Shuo Zhang and Krisztian Balog. 2020. Evaluating Conversational Recommender Systems via User Simulation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '20)*. 1512–1520. https://doi.org/10.1145/3394486.3403202

Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. 2018. Towards Conversational Search and Recommendation: System Ask, User Respond. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18)*. 177–186. https://doi.org/10.1145/3269206.3271776

Yinan Zhang, Xueqing Liu, and ChengXiang Zhai. 2017. Information Retrieval Evaluation as Search Simulation: A General Formal Framework for IR Evaluation. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval (ICTIR '17)*. 193–200. https://doi.org/10.1145/3121050.3121070

Yinan Zhang and Chengxiang Zhai. 2015. Information retrieval as card playing: A formal model for optimizing interactive retrieval interface. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 685–694.