

# Task-completion Engines: A Vision with a Plan

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

## ABSTRACT

This paper presents a vision and a plan for task-completion engines that support humans in solving complex, knowledge-intensive tasks, by providing an integrated environment that caters for all task-related activities. We propose three specific use-cases, describe the desired functionality from the users' perspective, outline the main components of such a system, and discuss evaluation methodology. We conclude by formulating next steps needed for making this vision become a reality.

## Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: H.3.3 Information Search and Retrieval

## Keywords

task-completion engines, complex search tasks, living labs

## 1. INTRODUCTION

Online information is a key notion in today's society. People turn to web search engines for a variety of reasons: to find products or services, to research medical issues, to seek new career opportunities, to plan a family vacation, to find the one true love, and the list continues endlessly. Along with the increased usage and audience came increased expectations regarding the search experience. Web search is currently undergoing a major paradigm shift, away from returning merely a ranked list of documents ("10 blue links") towards more explicit and focused responses. For example, when querying for "Chinese restaurants in Brussels," restaurants are displayed on the city map of Brussels, when asking for the "weather in New York," a 7-day weather forecast is shown, searching for "books by Stephen King" returns the book covers, titles, and publication dates, along with information about the author, and the question "How high is the mount everest?" is answered by "8,848 meters." Users expect the system to "understand" the intent and meaning behind the search query, consider the context (such as location and time of day), and respond to it directly and appropriately; thus, search engines are transforming into *answering engines*.

Search, however, is rarely performed for its own sake, but is usually associated with a specific target or goal. In many cases, this goal is the completion of a larger task, which is often complex (involving a nontrivial sequence of steps) and knowledge-intensive (requiring access to and manipulation of large quantities of information). Planning a family vacation or setting up a task force are just two of a plethora of examples. Such tasks call for a potentially large number of search queries to be issued in order to collect all the information needed. Indeed, it has been estimated that these "research missions" account for 10% of users' sessions and more than 25% of all query volume [7]. It has also been shown that almost 60% of complex information gathering tasks are continued across sessions [8]. Yet, completing a task requires more than just search. The gathered information needs to be manually curated, with gaps filled in where necessary. And, it often takes additional data processing steps (filtering, sorting, aggregation) before an actionable decision can be reached. Contemporary search environments are tailored to support a small set of basic search tasks and provide limited help in this tedious process. Resolving complex tasks with current search technology often requires us to use multiple search sessions and multiple search strategies, and then manually synthesize and integrate information across sessions (e.g., by opening multiple windows or tabs and cutting-and-pasting information between them). To solve these problems, one needs a paradigm shift from answering engines to *task-completion engines*.<sup>1</sup>

One the high-level, such task-completion engines

- can provide intelligent support and assistance, both for routine procedures and new tasks;
- can offer systematic production of data that is verifiably attributable to its source;
- can perform logical reasoning over knowledge (as opposed to mere statistical operations on words);
- are able to learn from user interactions and ultimately generalize to arbitrary tasks;
- are intuitive, easy-to-use, and shield the user from the complexities of the underlying processes.

### 1.1 Prior art

People engage in a wide variety of interactions with information; web search engines is only one among many (albeit one of the most important). Much of the research in this area has focused on detecting when a user is embarking on a (potentially) long search task using implicit interactions [1] and trying to address people's information needs directly within web search result pages [4]. We take a

<sup>1</sup>Note that we use the term task-completion in a restricted sense, related to "knowledge work." Importantly, we are not concerned with managing tasks that linger on a person's todo-list nor with creating workflows or actions plans.

different stand; instead of extending contemporary web search engines with task-based support, an approach that is inherently limited in nature and scope, we re-think the whole search experience.

Task-based search from an information seeking perspective has generated several notable task models and documented how task type and task properties can impact search behaviour [12]. E.g., it has been shown that the increase of task complexity increased the complexity of information and the number of sources needed, but decreased the success [3]. Ruthven [9] presents an overview of interactive IR systems and highlights how little we know about the mechanics of interaction during a process of performing a complex task. In responding to this need, Toms et al. [10] have explored the boundaries of the work task and search process to examine how users integrate search with the larger task, and found that two-thirds of time spent on the task was spent after finding a relevant set of documents. They conclude that “the ultimate challenge will be in building useful systems that aid the user in extracting, interpreting and analysing information to achieve work task completion” [10].

## 2. USE-CASES

We present three use-cases below as examples of complex, knowledge-intensive tasks. These specific use-cases are selected because they are of interest to a broad audience, while being sufficiently diverse from one another to be able to exhibit behaviour that is peculiar to each of them. Also, there is a great amount of related work available for each of these application domains (concerning data, query understanding, and retrieval) that can be capitalized upon.

**Travel planning** is chosen because it does not require much of an explanation; people already use a number of websites, apps, and services that address various travel-related issues. This helps us in formulating an initial set of requirements both in terms of function and content. Also, in the words of the Travelstormer site,<sup>2</sup> “travel planning is 90% decision-making.”

**Shopping** is one of the main online activities. A great variety of tasks is performed, from simple price comparison (“where to buy X”), to researching and comparing products (“find me products similar to X”), to more involved scenarios that evolve and develop over longer periods of time (e.g., buying a house).

**Setting up a work force** is about forming a group of people and/or organizations (often with complementary skills) that together can accomplish a larger task. Real-world examples include finding contractors to renovate a house and setting up a committee or a project team.

## 3. USER INTERFACE AND INTERACTION

Let us consider travel planning as our use-case and imagine a family planning a vacation to Croatia. Due to space constraints, we cannot provide a detailed cognitive walkthrough. We, however, can easily list a number of information needs in this context:

- *How to get there?* Should we fly and rent a car (or scooters) there or should we drive?
- *Where to stay?* Can we bring pets? Is there a discount for children? What about parking possibilities?
- *What to do?* Are there any activities at that time? Are there beaches nearby? Are those beaches suitable for children?
- *How much will it all cost?*

<sup>2</sup><http://travelstormer.com>

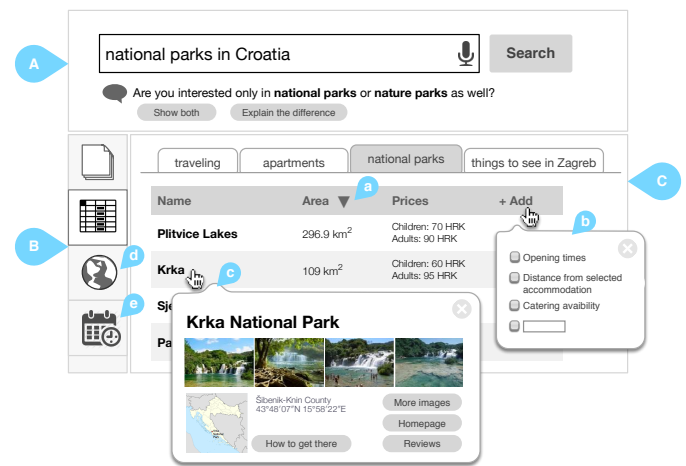


Figure 1: Excerpt from the envisaged user interface.

Finally, when all this information has been collected, the family might ask the question before deciding: *What if we go a week later?*

Our goal is to provide an intuitive, easy-to-use interface, based on elements that people are already familiar with. The envisaged UI is a combination of the single-search-box paradigm (with a possibility for voice input), spreadsheets, and conversational interfaces, put simply, “Google-meets-Excel-meets-SIRI.” Figure 1 shows an excerpt, where three main areas are highlighted: (A) the conversational search interface; (B) views over the data; (C) data under the selected view. The illustration displays the “tabular” view, where all information related to the task is presented in spreadsheet tabs; data can be sorted (a), filtered, or complemented with additional columns (b). Upon selecting an entity, further information can be obtained (c). Custom views (i.e., beyond the tabular display) are also available, depending on the particular task or task stage, for example, based on location (d) or time (e).

## 4. KEY COMPONENTS

The previous section presented the envisaged system from the user’s point of view. Next, we discuss the main components from a system perspective; these are shown on Figure 2.

**Task modeling.** Tasks differ across a number of attributes and can be characterized as a function of the task structure, content (types of entities involved), user, and/or the user’s context. The key lies in understanding the workflow of the users and distilling it into the correct discrete steps. We view tasks as being made up of smaller sub-tasks or components, with “information requests” (see below) being the atomic units. A framework is needed for modeling the transitions between task stages, including the probabilities of transitions, and transition triggers (information requests); see, e.g., [11] for a possible approach based on Markov models.

**Semantic analysis.** We designate semantic analysis as a separate building block that is shared across the three components that follow next. It comprises of methods and tools for identifying entities, attributes, and relationships, in different types and sources of data (both unstructured to structured), with cross-linking and contextualization within the scope of relevant tasks and interactions.

**Request modeling and understanding.** We use the term *information request* to describe any action or operation performed by the user. These information requests may come in different flavors,

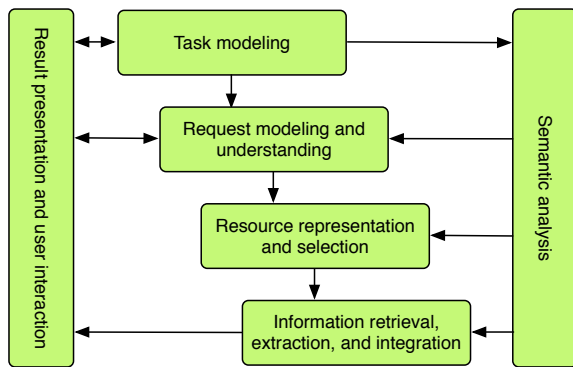


Figure 2: Key components and their dependencies.

for example, issuing a keyword query, sorting a column, selecting a webpage for examination, etc. Information requests serve as our atomic units for task modelling. Requests that require the retrieval of new information are satisfied by first selecting appropriate sources, then retrieving and extracting information from one or more sources (through processes independent of each other), and finally combining information from multiple sources and using probabilistic inference to arrive at the final results.

*Resource representation and selection.* This component is concerned with the identification of data sources that potentially contain valuable information in the context of a given task. We distinguish between two main types of data sources: unstructured and structured. The former is the *document web*, which can be accessed through (the APIs of) major web search engines. The latter is the *Web of Data*, comprised primarily of Linked Data resources, but is not limited to open sources. Standard protocols, such as RESTful Web Services, combined with authorization mechanisms, like OAuth, make it possible to provide access to confidential data that must not be exposed to the Internet directly.

*Information retrieval, extraction, and integration.* This module deals with the extraction, ranking, and fusion of information from the multiple sources of evidence. We consider entities as key information units for organizing information and strive for a structured entity representation. In unstructured sources, such representations may be obtained by first identifying vital documents and then extracting entity-related information from them. It is important to keep provenance information for all results.

## 5. EVALUATION METHODOLOGY

Evaluation needs to be performed both for the end-to-end solution and for the individual components.

*End-to-end evaluation.* In evaluating the overall usefulness of the system we consider the engine’s ability to help the user accomplish a task from start to finish. This poses significant challenges, mainly because of the inherent non-replicability. We address this by implementing the envisaged system as a public demonstrator that operates as a living lab platform.

*Component-level evaluation.* Component-based evaluation can be performed using both community-based evaluation exercises and studies around the specific use-cases using data collected with the public demonstrator. Specifically,

- *Semantic analysis* can use evaluation methodology and benchmarking frameworks developed for entity linking, e.g., [5]. Further evaluation resources include the recently released Yahoo search query log to entities (Webscope dataset L24<sup>3</sup>).
- *Request modeling and understanding* requires purpose-built evaluation resources, addressing the task-specific aspects; it can be established from actual usage log and click data, collected by the demonstrator. The upcoming TREC Tasks track aims to evaluate a system’s ability of understanding the set of possible tasks a user is trying to achieve given a query.
- *Resource representation and selection* can be evaluated using the TREC Federated Web Search track [6].
- *Information retrieval, extraction, and integration* can make use of the TREC Knowledge Base Acceleration track’s platform for the extraction and integration part; for core entity retrieval, a test collection based on DBpedia is provided in [2].

## 6. CONCLUSIONS AND NEXT STEPS

In this paper, we have presented a plan for a task-completion engine that supports humans in solving complex, knowledge-intensive tasks, by providing an integrated environment that caters for all task-related activities (which, to date, are performed using a combination of various tools, applications, and services). Specifically, we have proposed three use-cases, described the desired functionality from the users’ perspective, outlined the main components of the system, and discussed evaluation methodology.

The road to operationalizing the envisaged system is long and fraught with technical obstacles and research challenges. A key to success will be making sure that this is a community effort as opposed to an individual (or small group) effort. The public demonstrator could serve here as a common platform that supports both development and in-situ evaluation.

## 7. REFERENCES

- [1] E. Agichtein, R. W. White, S. T. Dumais, and P. N. Bennet. Search, interrupted: understanding and predicting search task continuation. In *Proc. of SIGIR’12*, 2012.
- [2] K. Balog and R. Neumayer. A Test Collection for Entity Search in DBpedia. In *Proc. of SIGIR’13*, 2013.
- [3] K. Byström. Information and Information Sources in Tasks of Varying Complexity. *J. Am. Soc. Inf. Sci. Technol.*, 53(7):581–591, 2002.
- [4] L. B. Chilton and J. Teevan. Addressing People’s Information Needs Directly in a Web Search Result Page. In *Proc. of WWW’11*, 2011.
- [5] M. Cornolti, P. Ferragina, and M. Ciaramita. A Framework for Benchmarking Entity-annotation Systems. In *Proc. of WWW’13*, 2013.
- [6] T. Demeester, D. Trieschnigg, D. Nguyen, and D. Hiemstra. Overview of the TREC 2013 Federated Web Search Track. In *Proc. of TREC’13*, 2014.
- [7] D. Donato, F. Bonchi, T. Chi, and Y. Maarek. Do you want to take notes?: identifying research missions in Yahoo! search pad. In *Proc. of WWW’10*, 2010.
- [8] B. Ma Kay and C. Watters. Exploring Multi-session Web Tasks. In *Proc. of CHI’08*, 2008.
- [9] I. Ruthven. Interactive information retrieval. *Annual Rev. Info. Sci. & Technol.*, 42(1):43–91, 2008.
- [10] E. G. Toms, R. Villa, and L. McCay-Peet. How is a search system used in work task completion? *Journal of Information Science*, 39(1):15–25, 2013.
- [11] V. T. Tran and N. Fuhr. Markov Modeling for User Interaction in Retrieval. In *Proc. of MUBE’13*, 2013.
- [12] P. Vakkari. Task-based information searching. *Annual Rev. Info. Sci. & Technol.*, 37(1):413–464, 2003.

<sup>3</sup><http://webscope.sandbox.yahoo.com/catalog.php?datatype=1>