Interaction for Information Discovery Empowering Information Consumers

Kurt Englmeier[1(✉)] and Fionn Murtagh[2]

[1] Schmalkalden University of Applied Science, Schmalkalden, Germany
kurtenglmeier@acm.org
[2] University of Derby, Derby, UK
fmurtagh@acm.org

Abstract. Information Discovery (ID) is predominantly addressed by approaches from Artificial Intelligence (AI). Automatic ID scans large amounts of data and identifies as many potential candidates for discovery as possible. Mass discovery may in fact serve the needs of many information consumers. However, that does not mean that it addresses a broad range of user interests, too. Economies of scale urge the development of automatic tools to address user needs only from a certain critical mass. Hence, many user needs remain unaddressed. This is where HCI comes into play and provides fundamentals for pattern languages that empower information consumers to stage their own information discovery. With this paper we want to draw attention to an approach that is developed around the paradigm of human-centered interaction design. We present an Open Discovery Language that can completely be controlled by information consumers.

Keywords: Information Discovery · Information extraction · Data science · Collaborative work · Interaction design · Participatory design · Pattern languages · Human-centred information management

1 Introduction

A more active role of the information consumers can be enabled by self-service features. This kind of self-service IT may point to user-friendly versions of analytic tools, enabling information consumers to conduct their own analytics. The smooth integration of domain and tool knowledge completes the picture of self-service discovery. The user experience is an important design paradigm, for search engines too. If the users have more impact on the retrieval behaviour of their search engines that leads to an appealing user experience, they are more willing to engage themselves in developing suitable scenarios for their search strategies [1]. When their needs drive design, information discovery engines will provide the insights they require. It is too often the case, that technology as required and described by the users is not quite well understood by designers. There are several ways of human-centred design to overcome this lack of mutual understanding. There is user-centred design where users are employed to test and verify the usability of the system. Participatory design [2] outreaches this form of user engagement, it understands users as part of the design team.

S. Yamamoto (Ed.): HIMI 2016, Part I, LNCS 9734, pp. 252–262, 2016.
DOI: 10.1007/978-3-319-40349-6_24

There are many retrieval methods and technologies that help us to discover the information we need. In general, they serve to detect prominent data in streams of structured or unstructured data. These data are prominent because they are meaningful to us, that is, they reflect information that satisfies our information need. They are also prominent because they follow certain patterns. Data patterns can be quite different, a particular sequence of characters in a string sent by a weather station indicates its location, temperature, and humidity, among others. In texts, an asterisk followed by a date usually indicates a birthdate. Data mining, much like information retrieval, identifies such patterns in structured or unstructured data. With Big Data the focus shifted towards mining of unstructured data. In particular sensors provide their data in unstructured form. The phenomenon, however, to mine unstructured data is not new. Big Data just adds volume, velocity, and variety to it. It addresses the necessity to handle high volumes of data covering a broader spectrum of information and continuously produced by all kind of devices. The Internet-of-things supports the ubiquitous connectivity of these devices and the exchange of data between them.

What qualifies for a prominent or meaningful pattern depends on the logic of the data mining or retrieval algorithms capturing and analyzing the data. The designers of mining features exclusively define this logic and thus the semantics that turn data into information. They determine which data can be meaningful to us. It is the designer, data scientist, or programmer that brings meaning into data, not we. Everything beyond their designs is out of reach for our information needs. What these specialists do not consider in their designs is simply not searchable. That holds for Web search engines in general, but also for individual data collections on the corporate level. Nevertheless, search engines enable us to find many interesting things and valuable information on the web. However, tailoring search engines to individual needs requires those specialists and is thus prohibitively expensive. This results in a search space not addressed by search engines. Probably, this non-addressed search space does not harbor any valuable information. However, there are reasons to believe that the contrary is true. New emerging disciplines like sentiment analysis show that there are still many things left to mine. Broadening the scope of search, however, means more variety in the design of search engines. Even though we can expect that progress in data mining and information retrieval will yield new emerging search engines that are more powerful and address more user needs, the non-addressed search space will remain significantly big.

To raise our search experiences we propose a search interface that supports the active role of the users: By lowering the technical entry level we empower them to equip search engines with their own search features. We want to enable users to define not only keywords but also essential qualities the retrieved results must have. These qualities cannot be expressed by keywords alone.

To reflect essential qualities in a query we propose to define keywords in combination with descriptive patterns rendering the qualities. As we will see in the next section, these patterns can be essential to retrieve the information we require or to discover the data that correspond to our information need. Our approach is applicable to unstructured information, but we can explain it better in the realm of text information. In Big Data, text is the most prominent type of data anyway.

2 Human-Centered Information Discovery

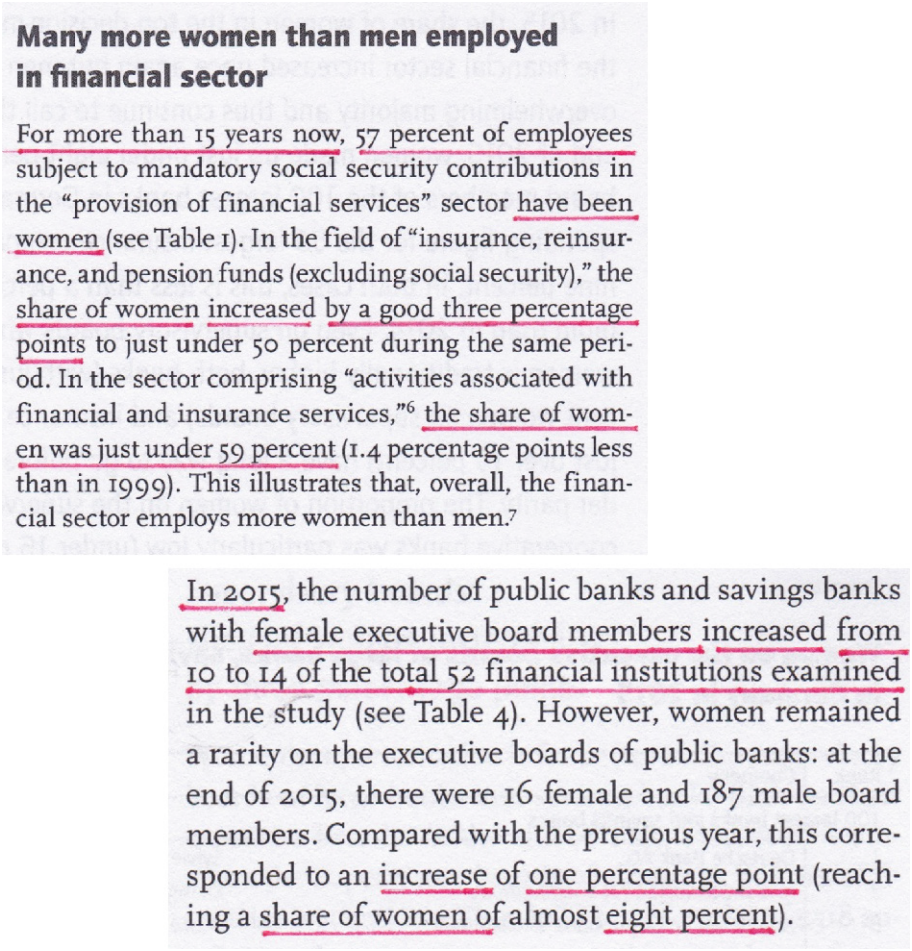
Information discovery (ID) broadly focuses on identifying semantic correlations among data that stand for a superior concept or meaning, not explicitly expressed by these data. Close collocation of data is often used as an indicator for a meaningful correlation. Mass discovery automatically locates frequently collocated data with the assumption that the findings can be useful to users. We call this process shallow discovery.

If we consider the question "Did the share of women in high-level positions of companies increase in the recent past?" we may get a series of articles when searching the web and using the terms of the question as keywords. By expanding and refining our query terms we may clean the query results from irrelevant documents. The articles retrieved may help us to answer the question. For this purpose, we have to read each text and check every section that may relate to our question. It may turn out, that a text addresses the topic, but reflects the corresponding situation in the 60s or 70s of the past century. To further refine our retrieval results, that is, to fine-tune them in accordance with our information need, we have to scrutinize the provided results. The document collection provided in the first place represents a shallow knowledge of the topic. We consider the more detailed and more relevant sections within the documents as deep knowledge.

The search engine can handle the shallow knowledge the web has on the topic, that is, the index list and stored queries related to the topic. Usually, this collection of documents is the first step in our search for documents that address our information need. It just summarizes our need that may have further ramifications not explicitly stated in the query. There are many reasons for this first vague query representation. One reason is that we cannot completely translate our information need into suitable query terms. If we are interested in the actual situation of women filling senior management positions, we may have a variety of aspects in mind that are hard to express in query terms. "Actual", for instance, may mean 2016, 2015, the past five years, or the most recent decade. However, if we expand our query by all terms reflecting the recent past, we are going to raise drastically the quantity of irrelevant retrieval results.

Implicit concepts, we are interested in, may cover "increase" or "number of positions filled", that is, statistical indicators that answer our query. The keyword "increase", for instance, may not help much, albeit it's an essential concept if we want to know whether a positive effect has occurred in the employment situation of women. We need to further specify the increase we have in mind. It may be increase in absolute or relative figures, but the increase should reflect the increase in number of positions (high-level position) filled by women. It may also address the increased number of companies with women in decision-making bodies. The increased number of men in these bodies may even indicate the contrary to what we link at first with the keyword.

Figure 1 shows the variations of the concept "increase" we are interested in. There are a number of instances of the concept "increase": "increase by 14 %", "increase from 10 to 14", "increased by a good three percentage points", or "virtually no increase over 2014". Furthermore, increases vary over industries and economic sectors, too. To correctly handle the concept "increase" we need further qualifying data that may be



Many more women than men employed in financial sector

For more than 15 years now, 57 percent of employees subject to mandatory social security contributions in the "provision of financial services" sector have been women (see Table 1). In the field of "insurance, reinsurance, and pension funds (excluding social security)," the share of women increased by a good three percentage points to just under 50 percent during the same period. In the sector comprising "activities associated with financial and insurance services,"[6] the share of women was just under 59 percent (1.4 percentage points less than in 1999). This illustrates that, overall, the financial sector employs more women than men.[7]

In 2015, the number of public banks and savings banks with female executive board members increased from 10 to 14 of the total 52 financial institutions examined in the study (see Table 4). However, women remained a rarity on the executive boards of public banks: at the end of 2015, there were 16 female and 187 male board members. Compared with the previous year, this corresponded to an increase of one percentage point (reaching a share of women of almost eight percent).

Fig. 1. Examples of deep knowledge: the variety of presentations of the concept "increase" (of the share of women in high-level positions) [3].

additional keywords, figures, and special characters. These qualifying attributes constitute the deep knowledge about the concept increase.

3 Participatory Design for Self-service Discovery

Even sophisticated search engines can only focus on mass needs for information. To enhance the users' search experience and to avoid confronting the users with too many irrelevant retrieval results, they support users by recommending useful query terms that may detail their need. Recommending keywords, however, only works if the system already identified a critical mass of query term sets that suit the user's keywords

expressed so far. If the user's query is more unique, the strength of the search engine diminishes. When an information need is no longer a mass phenomenon the users are on their own, that is, they have to continue their search manually.

Information consumers can usually sketch their information demand that summarizes the data they need to solve their information problem. They have a deep understanding of the foundations of their domain. The integration of more competence, in particular domain competence, can lead to a more active role of the human actor for her or his own discovery experience [4]. This kind of user engagement goes beyond user requirement analysis, participatory design, and acceptance testing during the development of discovery features.

People working with information have a data-driven mindset [5, 6], that is, they resort to mental models [7] that abstractly reflect the information they expect to encounter in their retrieval space [8]. This mindset enables them to sketch blueprints of the things they are looking for, that is, blueprints of their information need. This conceptualization of an information need is reflected in its abstract representation. It has three important qualities: The concept of the information need can be communicated, collectively refined and operationalized. Conceptualization means that users can sketch scenarios showing how their information need may manifest in data, that is, what kind of data patterns it may take. Discussing these scenarios on group level leads to more reflected scenarios and thus to more sound concepts of an information need [9].

Experimenting with data is an essential attitude that stimulates data discovery experiences. People initiate and control discovery by a certain belief - predisposition or bias reinforced by years of expertise - and gradually refine this belief. They gather data, try their hypotheses in a sandbox first and check the results against their blueprints, and then, after sufficient iterations, they operationalize their findings in their individual world and then discuss them with their colleagues. After having thoroughly tested their

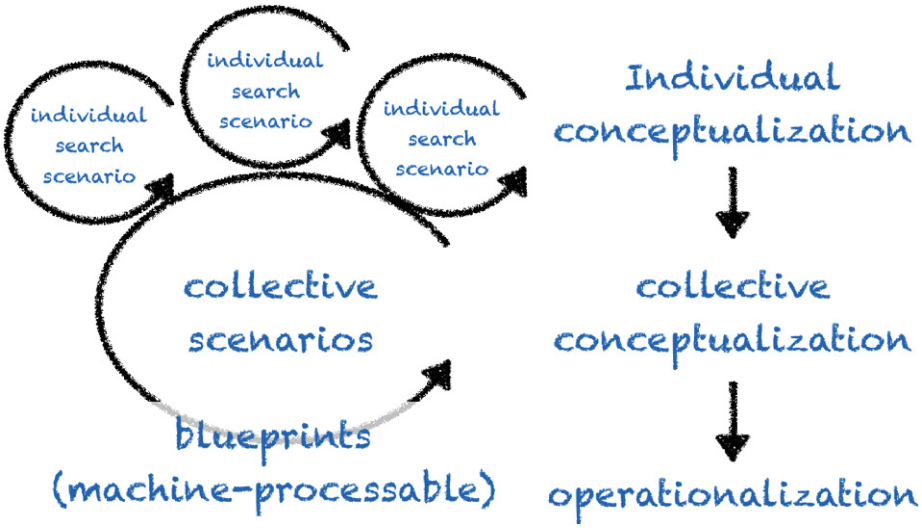


Fig. 2. Discovery lifecycle from individual conceptualization to operationalization.

hypotheses, information consumers institutionalize them to their corporate world, that is, cultivate them in their information ecosystem.

Much like conceptualization, the operationalization of an information need can be an individual or a collective task. Information consumers can express their search scenarios in a way that later on can be processed by machines. These blueprints of search are far from being programming instructions but reflect the users' "natural" engineering knowledge. The machine then takes the blueprints and identifies these information scenarios in data, even though the blueprint abstracts away many details. The language knowledge and their mental models constitute shallow knowledge necessary and sufficient to engineer statements that are processable by the search engine [10, 11]. After reflecting the corporate blueprint they may get hints for further discoveries and the participatory cycle starts anew (see Fig. 2).

4 Pattern Language for Discovery

A good starting point therefore is the language of information consumers. To avoid any irritations or ambiguities, people try to be quite precise in their descriptions. Even though these descriptions are composed of natural language terms and statements, humans are quite good in safeguarding literal meaning in their descriptions. For us, the data-driven mindset becomes evident when users can turn their domain and shallow engineering knowledge into machine instructions suitable for the precise detection and extraction of the facts they expect to discover. The users are in the position to express their domain knowledge on a certain level of abstraction. If the technological entry level for tagging their domain knowledge suits the users' engineering knowledge we can expect them to develop operable representations of their information need [12].

4.1 Design Principles

The blueprint of an information need serves two purposes: they reflect semantic qualities of the discovery scenario the search engine shall detect in data. Simultaneously, they are the building blocks of the meta-language that, when correctly syndicated, support data integration and sharing. While syndicating metadata along their domain competence, users foster implicitly active compliance with organizational data governance policies.

Information discovery starts with information extraction (IE) [13, 14] that distills text or even scattered documents to a germ of the original raw material. IT experts engineer information extraction systems that operate on templates for the facts to be extracted. Labelled slots constitute these templates whereby the labels represent annotated terms.

Self-service information discovery starts with user-driven IE. The users first have to engineer their extraction templates that can also be considered as entity recognizers. This means, a certain amount of engineering is indispensable in IE. The key question is whether information consumers have the necessary engineering knowledge to handle discovery services on their own. This (shallow) engineering knowledge is assumed to

be acquired easily and thoroughly specific to the task at hand [15]. The assumption in self-service discovery is that users with their domain competence and shallow engineering knowledge are in the position to manage a certain level of data discovery and information sharing on their own.

The users' blueprints may be simple and concise, but they are comprehensive enough to cover their request. This, in turn, fosters the control of the discovery process. A template with, say eight to twelve slots, can comprehensively cover real-world requests in small-scale domains. This low level of complexity makes it easy for the information consumer to manually control discovery. Whenever they encounter unfilled slots or mistakenly filled slots they may check the corresponding document for obvious errors. On the other hand, they may adapt their blueprint if the representation of a fact appears to be sound in text, but the slots do not consistently correspond to the qualities of the fact.

Our discovery language, developed along the paradigm of simplicity [16], is a meta-language that serves the description of named entities on different levels of complexity. It enables users to define patterns for concepts rendered by simple entities like "date", "birthday", "address", or "increase" that gradually increase in complexity when hierarchically combined into larger constructs. The constituting elements (keywords and/or character patterns) of such entities are not necessarily juxtaposed in a text, in particular if we consider more complex concepts like "vendor" or "buyer" in a contract, that may consist of the named entities "name", "address", probably "birthday" or "social security number", "tax payer number", and "nationality". Furthermore, these elements may be scattered over a large section of the contract and do not always appear in the same sequence. There may be also text (or data) elements in between that are irrelevant for the user. These elements have or may take the character of stop words.

4.2 Language Syntax

The discovery language thus enables the definition of patterns that precisely reflect the information entities the users are looking for. The patterns help to locate relevant information by fading out irrelevant sections. The patterns may become quite complex. The limits of real estate, for instance, can be a combination of geographic directions, addresses, and street names. However, the users know the characteristics of the entities they are looking for. In turn, they can sketch an abstract representation or blueprint of these entities. Sketching means they describe what elements may appear together, in a strict or arbitrary order, over a narrow or broad range. They add significant leading or trailing elements if appropriate. The blueprints themselves can be used as elements and thus become building blocks of a more complex pattern.

On its lowest level, our discovery language completely works with Regular Expressions. They are a very useful instrument when it comes to identify patterns in text and in unstructured data, in general. Unfortunately, Regular Expressions are powerful, but absolutely not user-friendly. They require special skills and are not easy to handle, in particular, when they are addressing complex, i.e. real-world, patterns. Besides, Regular Expressions representing high level facts are extremely complex and barely manageable, even for professionals. Their application also has limitations, when

relevant elements (qualities) of facts are too dispersed over the data set, that means when too much "noise" appears between facts or their qualities.

We therefore propose a language that adopts Regular Expressions but shields users from their complexity. It provides users with a stock of labelled Regular Expressions addressing entities like "word", "tax payer number", "percentage", "phrase", "decimal" etc. Instead of defining Regular Expressions, the users compose their patterns by resorting to these basic patterns or to the patterns they have already defined by themselves. The syntax serves to describe how facts, as the constituent parts of the fact requested by the user, appear as sequences of word patterns in data. The corresponding descriptive pattern gradually aggregates into the complex pattern for the requested information. Internally it translates the users' instructions into complex Regular Expressions and applies it to the text or unstructured data.

As already pointed out, the language provides a set of basic variables that cover patterns like "date", "percentage", "zip code", etc.

The users take these basic language elements and define more complex patterns. The following expressions support the identification of companies in texts:

```
name,names = Words."Inc.";"company";"GmbH";"S.A."
street,streets = numeric.Words.ordinal;numeric."st."
city = Words.zipcode.Words:country
company,companies = name,street,suite,city
```

On the left side of the equation the user defines the name of the pattern. The right side of the equation lists the sequence of pattern elements. The pattern can be assigned to a singular term and optionally also to its corresponding plural term. When defining their own patterns, people intuitively apply both forms as you see in the first pattern ("Words").

The operators have the following functions:

- The dot indicates strong sequence ("followed by"). The element indicated before the dot must be located before the one indicated after the dot. In "street", there is a number followed by one or more words, followed by an ordinal or numeric expression that finally is followed by a keyword (put in quotes).
- Comma means weak sequence. The elements are expected to appear sequentially in the data but in any order. One or more elements may even be absent. The pattern "company" consists of the patterns "name", "street", "city" we defined before and the basic pattern "suite" (provided by the discovery language).
- The semicolon is used to indicate an exclusive combination. One of the elements must be present. The company name terminates with one (and only one) keyword as indicated in the example.
- A leading question mark indicates that an element or group of elements can be optional, that is, the corresponding pattern can but need not be located in the data.
- For the text examples of Sect. 2 (see Fig. 1) we can define some of the patterns as follows:

```
share = percentage."employee".industry sector."women";
"share of women";("female".Words).?percentage
share increase = share,"increase",percentage;range
```

The output of the extraction process is simply rendered in XML, in order to enable a smooth integration into follow-up processes for data analytics, visualization, reporting and the like. The slots (with annotated terms) capture the facts according to the users' pattern definitions render them by XML elements. Basic patterns are treated like primitive data types; entity elements that correspond to them are not explicitly tagged.

The extracted data for the example may appear in the following XML format:

```
<share increase>
    <share>
          female executive board members
    </share>
    <range>
          from 10 to 14
    </range>
</share increase>
```

In the end, we get a semantic representation of concepts that resembles a thesaurus. However, it is not a general thesaurus, rather an individual one ("share of women in high-level positions"), adapted to the specifics of the respective domain and information need. Furthermore, it isn't either a strict thesaurus where all its concepts are tightly integrated. It's rather a collection of more or less loosely coupled fractions of a thesaurus, with its fractions dynamically changing both, in their compositions and relationships among each other. It is thus more suitable to consider semantic badges as ingredients of a common vocabulary. This vocabulary, in turn, is the asset of the information consumers. They manage it in cooperative authorship.

First experiments with our discovery language addressed about 2500 documents (economic reports, real estate contracts with related certificates, diagnosis reports from radiology and product descriptions, to mention the most prominent ones) distributed over dozens of data sources. In the first place, these samples may seem small to validate a pattern language or to underpin its scalability. However, the inherent character of basically unstructured data distributed over different sources reflects the nature of the challenge we face in data discovery. The language applied in the documents addressed is quite uniform and not narratively complex. The document samples cover this language in its entirety, and thus scale for even larger collections. In many information ecosystems we barely have to deal with highly complex narrative forms. Due to this fact, we can consider this approach as scalable also towards thematic areas outside the domains addressed so far, as long as the narrative nature is relatively uniform.

5 Conclusion

Today's search engines are powerful. They changed the way we deal with information and turned into one of the most valuable information source we have. However, there is a chance and probably even a need to provide users with an appealing retrieval experience that reaches beyond what's possible today. To say, "Just formulate a query

and you will get thousands of interesting results! Take what you need!" reflects a strategy doomed to failure.

Information consumers want the information that exactly meets their particular needs. They expect a certain variety of information echoing the diversity of their information need and appreciate more meaningful information for their analytics, reporting, or whatsoever. The more retrieval systems meet the expectations of information consumers the higher are the advantages. These can be manifold, far beyond business success such as growing revenues and market shares.

This affects information governance in general and Big Data governance in particular. It should include semantic search and consider participatory design as design paradigm. There are many discovery tasks that serve individual, ad hoc, and transient purposes. Main stream discovery, in contrast, supports reoccurring discovery requests commonly shared by large user communities and operates on large data collection, including sometimes the entire Web. We can conceive manifold scenarios for non-mainstream discovery. Users may have to analyze from time to time dozens of failure descriptions or complaints, for instance. The corresponding data collections are personal or shared among small groups and consist of bunches of PDF files or emails, for instance, barely documents on the Web. Dynamically changing small-scale requests would mean permanent system adaptation, which is too intricate and too expensive in the majority of cases. With a flexible self-service solution like our discovery language information consumers can reap the benefits of data discovery and sharing and avoid the drawbacks of mainstream discovery.

The actual version of the our discovery language described here is available on sourceforge.net: http://sourceforge.net/projects/mydistiller/.

References

1. Shneiderman, B.: Designing for fun: how can we design user interfaces to be more fun? Interactions **11**(5), 48–50 (2004)
2. Robertson, T., Simonsen, J.: Challenges and opportunities in contemporary participatory design. Des. Issues **28**(3), 3–9 (2012)
3. Holst, E., Kirsch, A.: Financial sector: share of women on corporate boards increases slightly but men still call the shots. DIW Econ. Bull. **6**, 27–38 (2016)
4. Clement, A.: Computing at work: empowering action by "low-level users". Commun. ACM **37**(1), 52–64 (1994)
5. Pentland, A.: The data-driven society. Sci. Am. **309**(4), 64–69 (2013)
6. Viaene, S.: Data scientists aren't domain experts. IT Prof. **15**(6), 12–17 (2013)
7. Norman, D.: Some observations on mental models. In: Gentner, D., Stevens, A. (eds.) Mental Models. Lawrence Erlbaum, Hillsdale (1987)
8. Brandt, D.S., Uden, L.: Insight into mental models of novice internet searchers. Commun. ACM **46**(7), 133–136 (2003)
9. Elbeshausen, S., Womser-Hacker, C., Mandl, T.: Searcher heterogeneity in collaborative information seeking within the context of work tasks. In: Proceedings of the 5th Information Interaction in Context Symposium, pp. 327–329 (2014)

10. Sawyer, P., Rayson, P., Cosh, K.: Shallow knowledge as an aid to deep understanding in early phase requirements engineering. IEEE Trans. Softw. Eng. **31**(11), 969–981 (2005)
11. Chin, G., Rosson, M.B., Carroll, J.M.: Participatory analysis: shared development of requirements from scenarios. In: Proceedings of the ACM SIGCHI Conference on Human factors in Computing Systems, pp. 162–169 (1997)
12. Heymann, P., Paepcke, A., Garcia-Molina, H.: Tagging human knowledge. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, pp. 51–60 (2010)
13. Cowie, J., Lehnert, W.: Information extraction. Commun. ACM **39**(1), 80–91 (1996)
14. McCallum, A.: Information extraction: distilling structured data from unstructured text. ACM Queue – Soc. Comput. **3**(9), 48–57 (2005)
15. Fan, J., Kalyanpur, A., Gondek, D.C., Ferrucci, D.A.: Automatic knowledge extraction from documents. IBM J. Res. Dev. **56**(3.4), 5:1–5:10 (2012)
16. Magaria, T., Hinchey, M.: Simplicity in IT: the power of less. IEEE Comput. **46**(11), 23–25 (2013)