Designing User Studies for XML Retrieval *

Miro Lehtonen
Department of Computer Science
P. O. Box 68
FIN–00014 University of Helsinki
Finland
Miro.Lehtonen@cs.Helsinki.Fi

ABSTRACT

Ever since the research on XML retrieval started, we have seen little cooperation between the researchers and the system developers in the field. Consequently, some of the issues that seem fundamental to the researchers are trivial to the developers. For example, the existence of the real users of XML retrieval is rarely questioned by those who make money selling such software. As an attempt to bridge the gap between the far too separated parties, we discuss some issues that have been subject to user studies in recent years and suggest improvements for them.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Query formulation*; H.5.2 [**Information Interfaces and Presentation**]: User interfaces

General Terms

Design, Human Factors

Keywords

XML Retrieval, search engine, user study, user interface

1. INTRODUCTION

The seemingly long societal distance between the real users of XML search engines, the developers and vendors of such systems, and the researchers in the field — including the INEX community — shows in the research questions the latter are trying to investigate about the first group. The controversies rarely surface as the researchers conduct their self-designed user studies where users from their own academic circles use a self-designed or other experimental system for XML retrieval. It is counter-productive that the assumptions that originate in the experimental implementations have such a strong influence on the user studies including the user tasks, the test environment, and the interpretation of the test material. It has been unclear whether the results of the user studies generalise, and if they do, we do not seem to know *how* they generalise. For example, we have trouble proving that the assumptions and conclusions hold for more than one document collection or more than one system for XML retrieval. Therefore, we are challenged by the fact that our user studies rarely lead to results that have an impact outside the academic world.

This paper has been inspired by the numerous misunderstandings and the terminological ambiguity the author has had to deal with, coming from the XML side of the world and speaking XML, and once again, approaching the scientists and scholars of Information Retrieval, like a visitor in their home field. The purpose of the paper is to bring up the differences in view between those who see the big picture around issues related to IR and those who understand the essence of XML, and, ultimately, to help bring those two sides closer to each other.

As user studies are a major source of controversy, we present arguments for and against the questions we are studying about the users of XML retrieval. In Section 2, we consider the frequently asked question about user preferences: Are XML elements better than whole XML documents? Section 3 is a response to a recent analysis where the structural hints in queries were regarded. Examples of the users of XML retrieval, who were never really lost, are introduced in Section 4. The kind of issues that are considered relevant according to the author of this paper are discussed in Section 5, followed by concluding remarks in Section 6.

2. XML ELEMENTS AND DOCUMENTS

It is a common argument to the benefit of XML retrieval systems that, instead of whole documents, we may also see document fragments in the list of results [4], or, alternatively, that we may start our navigation from a relevant entry point in the result document in addition to the beginning of the document [5]. In order to emphasise this advantage, we seek support for element retrieval systems in user studies where users show interest either in whole XML documents, or single XML elements, or both [7]. Without the support, the entire need for element retrieval systems could be questioned[1]. In a general setting, however, "ele-

*SIGIR 2006 Workshop on XML Element Retrieval Methodology August 10, 2006, Seattle, Washington, USA.

[1]See the Call for Papers of this workshop.

ments or documents?" seems to be an irrelevant research question.

Firstly, the XML constructs such as elements and documents are part of the technical implementation of the document collection. The same content may be stored at various levels of granularity so that we have an XML document for each article, each journal, or even a whole volume of journals. When talking about Enterprise Content Management (ECM) and XML, the modern documentation tends to consist of much smaller units where a couple of paragraphs constitute a whole XML document. Consequently, the producers of the content do not even know what kind of publications (those too are documents) their content will be part of [2]. When it comes to the users of XML retrieval, they do not know, nor do they care, how much content a single XML element or an XML document holds. Given an arbitrary answer from the result list to some query, the user can hardly know whether they are inspecting a whole XML document or a part of an XML document. It is all just content to the user.

Secondly, when content is stored in the XML format, we always retrieve XML elements, either small ones or big ones. If the content is stored in an XML database, the retrieved answers can be whole documents, as well, though not necessarily the same ones as stored in the database. The most common logical units of XML that are returned include single XML elements, sequences of XML elements, and whole XML documents. Nevertheless, the concept of a "whole XML document" is rather useless for the research on IR as it is merely a technical detail. In order to straighten up the problem with the wording, we may want to say that the size of the answers in XML retrieval is dependent on the query instead of being fixed to the size of a document in traditional document retrieval.

As it makes no sense to ask the users whether they prefer XML elements to whole XML documents, at least according to the presented argumentation, we may still ask them if the size of the answers is "too small", "too big" or "just right" [6]. In the experiments of the Interactive Track of INEX, the users are expected to assess each answer's need of context by selecting one of the following values for each answer: Broad, Exact, Narrow (Task C). The context is defined as the content of the source document that is not included in the answer. The relevance of this question too can be disputed. In a realistic setting, the user may know nothing about the context, or even that there is one outside the returned XML element. If we are given a paragraph that is extracted from a scientific article, we cannot always tell that it actually comes from a whole article. When an answers seems too small to satisfy the information need, there is no guarantee that any bigger answer such as the "parent element" is somehow more appropriate to the user, but, knowing the context might just make the otherwise good answer look too small (Narrow Answer). The user assessment is thus unjustly biased by the choices made in the design of the user study.

3. STRUCTURAL CONSTRAINTS

XML is sometimes called a "metalanguage" because the document structure including XML elements is expected to describe the character content (text). What makes searching XML documents different — and more interesting — than searching plain text or hypertext documents is that the queries may include conditions on the document structure, as if we were querying a textual database. The structural conditions have been introduced to the IR researchers in the Content-And-Structure (CAS) queries of INEX in the past few years. XQuery [12] is a common query language that supports such conditions, but a simpler language called NEXI [11] is used in the context of INEX.

For a good reason, the structural conditions have been given the role of serving as hints rather than requirements for the search engine: they are not necessary or even very useful when the search engines evaluate the official queries of the past INEX initiatives. Trotman and Lalmas go even further in their interpretation, according to which, "structural hints in queries do not help XML retrieval" [10]. They do mention that this might not hold for arbitrary document collections, but is it even true of the single test collection? Questioning the value of the structural hints is justified, but generalising the claim to all the queries or all types of "structural hints" is not. Examples will follow.

Trotman and Lalmas also suggest that the users be particularly bad at giving structural hints. By this claim, in fact, they imply that it is possible to give such structural hints that help XML retrieval in the context of INEX and the IEEE collection. In this paper, we have more faith in the searchers as we point our finger towards the document structures as we present our claim: **Whether the structural hints help or not depends on the document type of the XML documents.** To be more exact, specifying structural constraints is useful and even necessary when the structure describes the content. However, the element names in the INEX IEEE collection of scientific journals do not describe the content, but instead, they describe the document structure such as paragraphs, sections, and article bodies. The structural hints given by the user thus describe the size of the answer they expect. Why specifying the constraints rarely helps is because users cannot know how much content is required to answer their query in the particular test collection. If an entire article describes the topic the user was interested in, we can hardly return a section to the user that would summarise the entire article. It is thus better to let search engines determine the best granularity of the relevant answers.

The Lonely Planet document collection of the INEX Multimedia Track[2] serves as an example of a case where the structural hints are useful. For example, someone who is interested in taking "cold showers" may want to specify that the keyphrase is not found in the `weather` element but in other elements such as `activities` (or why not `amenities`?) instead. Medical patient records in XML format [3] are another good example. If we want to know how to cure "fever", we want to see that keyword in the `diagnosis` element and exclude the occurrences in the `complaint` and `side-effect` elements. The relevant answers would most likely be `treatment` elements. In these examples, the most natural interpretation of the structural conditions is to treat

[2]These documents describing travel destinations are also known as the WorldGuide.

them as strict requirements, which further emphasises their importance.

Studying whether the structural hints help or not might be interesting, but it is rather unclear whether the results would have any impact in practical applications. We take the same attitude towards user studies investigating whether users can or cannot specify structural conditions in a specific query language such as NEXI. Whether any query language is too complex for the users is not a real issue, because the users and the query languages never meet each other. In practice, the structural hints are next to trivial for the users to specify as the search interfaces accommodate the procedure. The search conditions are typically defined with checkboxes, lists, and input fields, as shown in the example in Appendix A. Wildcards, logical operators, and regular expressions are naturally supported, but not required, as the input can be directly inserted into an XQuery expression. When the user interface is appropriately designed and when the structure of the documents is consistent, the users that were not able to specify good structural hints for the INEX queries can most likely give the exact "structural hints" that they need.

4. LOST & FOUND: THE USERS

Studying what users think of experimental systems for XML retrieval is likely to lead to experimental results, but nothing more, as the setting is often artificial. Experimental systems that index test documents rarely have a true demand that has originated in a user community. Studying user behaviour with real-life systems is thus considerably more reliable as a source of useful results. The biggest challenge to the INEX community, so far, seems to be the lack of contacts with the users of real-life element retrieval systems [9]. Although XML retrieval is a rather young field of research, the vendors of such systems have been happily selling their products for years. Moreover, the earliest user studies date all the way back to 2000 [13] which is two years before the first round of INEX. Rather than trying to find the users, we are tempted to ask an even more interesting question: How did we lose sight of the users of XML retrieval?

The answers are not simple. First of all, the search engines were not called systems for XML retrieval until quite recently. Secondly, XML does not have any bigger role in the systems than that of the document format. The users never have to see any XML markup when they use such systems. Consequently, most users of today's XML retrieval systems are not aware of being ones. Six years ago, the users liked their XML retrieval systems only because "it uses XML" [13]. By the time the concept of XML retrieval was well established, the users were no longer excited by hearing the three letters; they only expect to have access to the relevant content of their XML repository.

The third cause of confusion is in the definition of an XML (element) retrieval system. To some, including the author, any search engine that indexes XML documents and returns the content to the user falls in the category of XML retrieval systems. To others, it is enough that the systems gives users access to incomplete documents which they call (XML) elements. However, as such systems do not require any XML technology in their implementations, the latter conception is somewhat questionable. What most people seem to agree about is that XML retrieval systems let users give structural hints about the searched documents and the returned answers. Regardless of our definition, the systems for XML retrieval are widespread.

In addition to the XML search engines that are used in private companies and enterprises, there are a number of such systems online that are available to public use. A common feature in the real-life systems is that they are not general-purpose XML search engines, but they specialise in indexing and searching specific document collections. The search engine and the indexing methods are developed together with the document type[3] as they both are a part of the document management system. A brief list of vendors and off-the-shelf software products that come with element-level search capabilities is presented in Appendix B.

A quite recent example of an XML retrieval system was developed for the New England Journal of Medicine.[4] The online user interface lets us search the full-text of various journals as well as medical case records and educational material. Only by using the system, it is nearly impossible to know that one is searching XML documents. However, all the users are what we are looking for — *users of XML retrieval*.

Another example of an online XML search engine gives access to the letters of Dolley Madison.[5] The user may browse the collection, as well as search for text, search by time period, people, topic, and location. Again, it is impossible to see that we are searching XML documents, but in this case, it is mentioned on the main page.

The user interface presented in Appendix A shows how useful the Document Type Definition (DTD) can be in the UI design. The users need not understand XML, DTDs, or query languages to be able to formulate accurate queries on XML documents. How to conduct user studies on these users is a real challenge unless we want to redesign the tests starting from the user tasks and ending at the interpretation of the results.

5. RELEVANT QUESTIONS

So far, we have questioned a whole lot of issues that the contemporary user studies address. In order to make the criticism constructive, we regard which issues are relevant enough to deserve more attention in the future user studies.

5.1 Assessing the size of the returned answer

If we use a system that returns entry points to relevant documents, it is not meaningful to assess the size of the answer because whole documents are returned. Nonetheless, if the system returns answers that are extracted from the source documents, we are interested in how good the system is, in the user's opinion, at determining the correct granularity of the answers. The Interactive Track of INEX already

[3]An XML DTD or an XML Schema development usually goes hand-in-hand with the development of the document type.

[4]http://content.nejm.org/

[5]http://rotunda.upress.virginia.edu:8100/dmde/

includes the assessment of the size of the document component in their Task C [6]. However, they instruct the users to estimate the size in terms of the context (the source document), which is everything but a user-oriented question for a user study. If the answer's being "too small" (Narrow) or "just right" depends on the content outside the answer, we are assessing the performance of the system. How satisfied the users are with the given standalone-type answer is not dependent on the context of the answer in the source document. To conclude, we want to know how content the users are with the size of the returned answers, but we need to assess the quality in absolute terms which a function of the need for its context is not.

5.2 Opinions on the search interface

The user interfaces of the operational XML retrieval systems online typically have a web form which the searcher first fills in and then submits to the server. The number of input fields varies, as well as the number of selections made by default. The search interfaces directed to public use do not allow the users to access the XML documents through any XML query language, but is the low-level access even desirable, as long as the users have a way to specify the structural constraints discussed in Section 3? It is still unclear to the author of this paper, what kind of users and what kind of search tasks would benefit from a different kind of a user interface for entering the query. Anyway, it is sensible to study the user's opinion on a user interface, even if only to improve the general UI design.

Although the query forms look similar from one system to another, there are big differences in how the search results are presented to the user. In a similar fashion to web search engines, we are often given a list of links along with a summary or metadata about the answer. Each link anchor may come with multiple options and targets. The targets of the links may include extracts from the source documents as well as XML fragments of various sizes. Thanks to the inherent nature of XML, the answers are simple to convert into HTML or PDF. What kind of browsing interfaces are the most suitable for the result lists is an open question as well as an interesting topic for a user study.

5.3 Comparative studies

Although some may argue that scientific articles are atomic units of retrieval [9], we also have users who presumably prefer smaller answers to their queries [7]. Rather than testing users on a single system for XML retrieval that gives them a choice between XML elements and whole documents, it is more sensible to let the users try out two different systems: 1) a traditional search engine, and 2) an XML retrieval system. If the users prefer having a choice of entry points to not having more than one, we can conclude that this aspect of XML retrieval is meaningful. The same applies to XML-aware systems that return XML fragments to the user instead of whole documents.

When these tests are performed on the INEX IEEE collection, however, the results do not automatically generalise to different XML collections. For example, even if users preferred whole articles to single paragraphs, we could not draw similar conclusions concerning the Lonely Planet collection. It is more tempting to assume similar preferences about other scientific literature, anyway. Despite being a popular research question, whether users prefer sections or articles does not really have anything to do with XML. User studies investigating the issue do not even require systems for XML retrieval.

Another way to study the benefits of XML retrieval is to compare different collections with each other instead of comparing different systems. For example, we may let the users search both the original Wikipedia documents and the converted XML documents. In the case of the Wikipedia documents, we see many similarities in the two versions. For example, both document collections can be segmented into small document fragments. The segmentation of plain text may even result in more natural segments than those that follow the boundaries of XML elements. The major difference, in general, would be that the structure of the XML documents can be included in the queries, whereas, querying the structure of the non-XML documents is far less trivial. In the particular case of the Wikipedia documents, though, the XML structure is not very useful, and the benefits of XML might not be so great.

6. CONCLUSIONS

We have presented a whole lot of arguments in the hopes of improving the potential impact of user studies. One of the key points to remember is that a typical user of an XML retrieval system does not know when they are searching XML documents. Moreover, because there is no one-to-one correspondence between the traditional documents and XML documents, the users can hardly appreciate the benefits of only being shown the relevant parts of the XML documents. They do appreciate seeing relevant content, though, and they dislike being shown irrelevant content. Furthermore, the users, who now might seem quite ignorant, are not aware of being in the process of specifying structural constraints for their query when they fill in the fields of an advanced-looking search form. The last concern is the user's ability to judge the technology behind the implementation. A user study may show that users appreciate certain functionality that XML retrieval systems offer, but the very users could not possibly judge the details specific to the implementation, including the use of XML. All these issues should be taken into account when designing user-oriented user studies for XML retrieval.

In this paper, we have also learnt other little details about XML. The structure of XML documents was originally designed to serve as metadata about the content. Including the structure in the queries should help the search engine find the relevant answers only as long as the structure describes the content. We should also keep in mind, that, for any textual documents, XML is the enabling technology rather than a straight jacket posing limitations.

7. REFERENCES

[1] ASTM International. *E2210-02 Standard Specification for Guideline Elements Model (GEM)-Document Model for Clinical Practice Guidelines*, 2003.

[2] D. Clark. Rhetoric of present single-sourcing methodologies. In *SIGDOC '02: Proceedings of the 20th annual international conference on Computer*

documentation, pages 20–25, New York, NY, USA, 2002. ACM Press.

[3] J. S. Hooda, E. Dogdu, and R. Sunderraman. Health level-7 compliant clinical patient records system. In *SAC '04: Proceedings of the 2004 ACM symposium on Applied computing*, pages 259–263, New York, NY, USA, 2004. ACM Press.

[4] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson. XML retrieval: what to retrieve? In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 409–410, New York, NY, USA, 2003. ACM Press.

[5] M. Lalmas and J. Reid. Automatic identification of best entry points for focused structured document retrieval. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 540–543, New York, NY, USA, 2003. ACM Press.

[6] B. Larsen, S. Malik, and A. Tombros. The interactive Track at INEX 2005. In *INEX*, volume 3977 of *Lecture Notes in Computer Science*, pages 398–410, 2006.

[7] B. Larsen, A. Tombros, and S. Malik. Is XML Retrieval Meaningful to Users? Searcher Preferences for Full Documents vs. Elements. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, Aug. 2006. To appear.

[8] R. Shiffman, G. Michel, A. Essaihi, and E. Thornquist. Bridging the guideline implementation gap: a systematic, document-centered approach to guideline implementation. *Journal of American Medical Information Association*, 11(5):418–426, Sep-Oct 2004.

[9] A. Trotman. Wanted: Element Retrieval Users. In *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology*. Department of Computer Science, University of Otago, July 2005.

[10] A. Trotman and M. Lalmas. Why Structural Hints in Queries do not Help XML-Retrieval. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, Aug. 2006. To appear.

[11] A. Trotman and B. Sigurbjörnsson. Narrowed Extended XPath I (NEXI). In *Advances in XML Information Retrieval, Third International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2004), Dagstuhl Castle, Germany, 6–8 December 2004, Revised Selected Papers*, volume 3493 of *Lecture Notes in Computer Science*, pages 16–40. Springer, 2005.

[12] W3C. *XQuery 1.0: An XML Query Language, W3C Candidate Recommendation*, 3 November 2005. Available at `http://www.w3.org/TR/xquery/`.

[13] L. Weitzman, S. E. Dean, D. Meliksetian, K. Gupta, N. Zhou, and J. Wu. Transforming the content management process at ibm.com. In *CHI '02: Case studies of the CHI2002 / AIGA Experience Design FORUM*, pages 1–15, New York, NY, USA, Apr. 2002. ACM Press.

APPENDIX

A. AN OPERATIONAL USER INTERFACE

If guidelines for clinical practice were stored in a plain text format, finding relevant information would require highly sophisticated methods for Information Retrieval. Thanks to the metadata provided by the XML format, we can easily make the queries so accurate that simple term weighting methods are sufficient. Figure 1 shows the beginning of a user interface of such system for entering search terms for the query.

Besides search terms, the structure of the indexed documents can be taken advantage of. Figure 2 shows how simple it can be — for the particular document type. User interfaces for searching other kind of documents should be modified accordingly to be functional.

More multiple choices are shown in Figure 3. From these screenshots we can see that the content producers may include quite a lot of metadata about the guidelines they describe. The DTD of the collection is public [1, 8] and also available online[6].

B. VENDORS

Table 1 shows a non-comprehensive list of vendors providing support for XML Element Retrieval.

[6] http://www.astm.org/

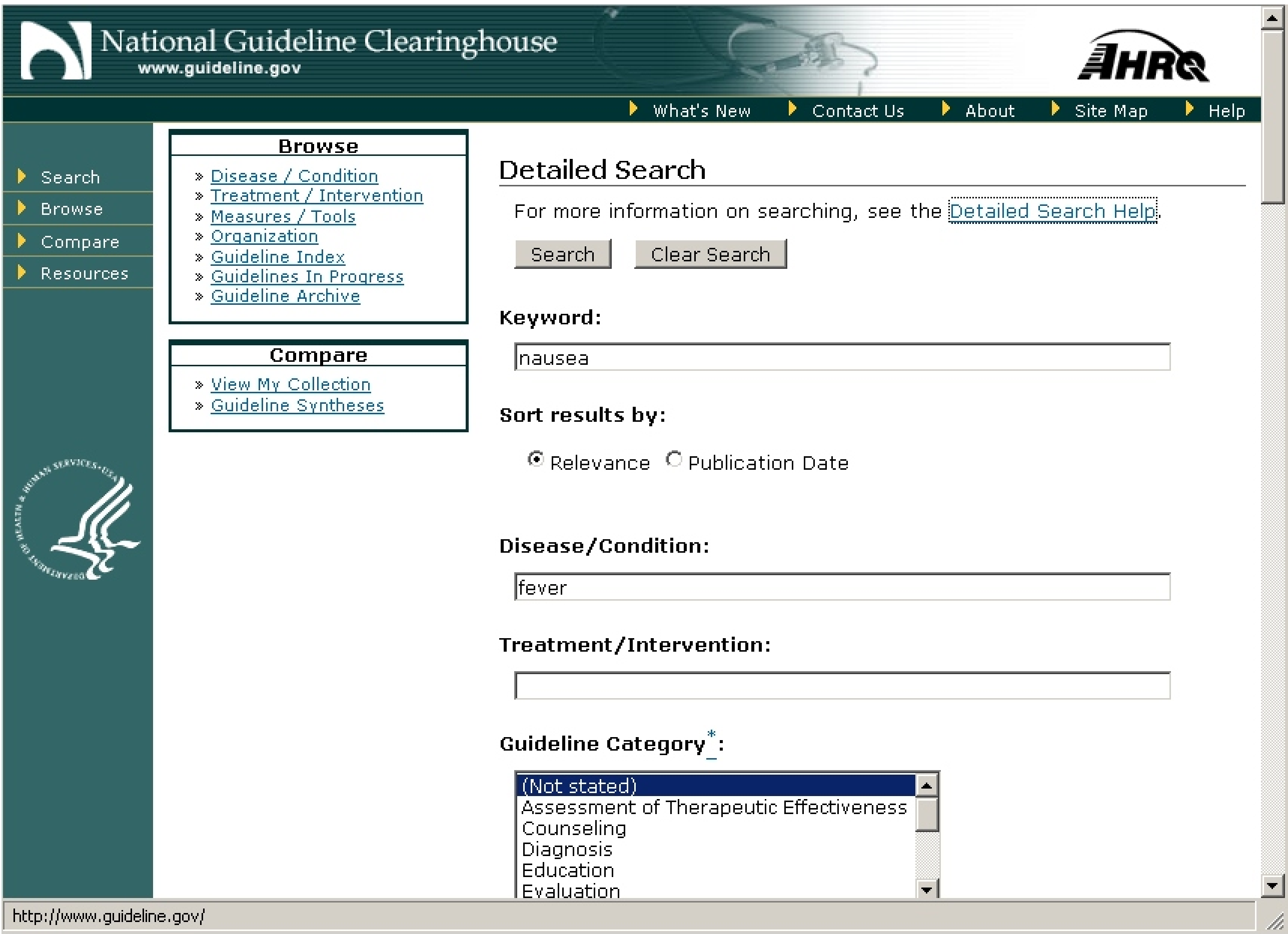

Figure 1: Input fields for entering keywords.

Vendor	Product	URL
Astoria Software	Astoria XML Content Management Platform	www.astoriasoftware.com
IBM	WebSphere Information Integrator OmniFind Edition	www.ibm.com
IXIASOFT	TEXTML Server	www.ixiasoft.com
Mark Logic Corp.	MarkLogic Server	www.marklogic.com

Table 1: Companies providing XML search engines followed by the product name.

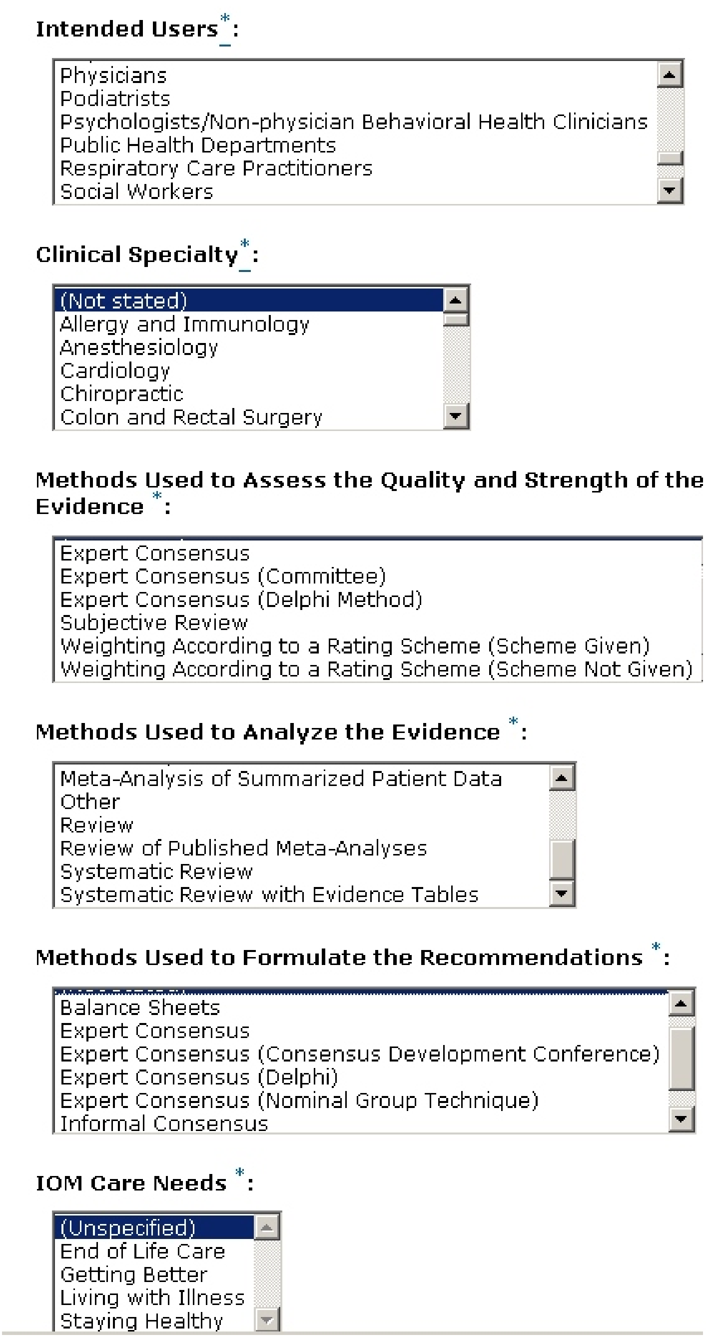

Figure 2: Lists helping users specify structural constraints for their query.

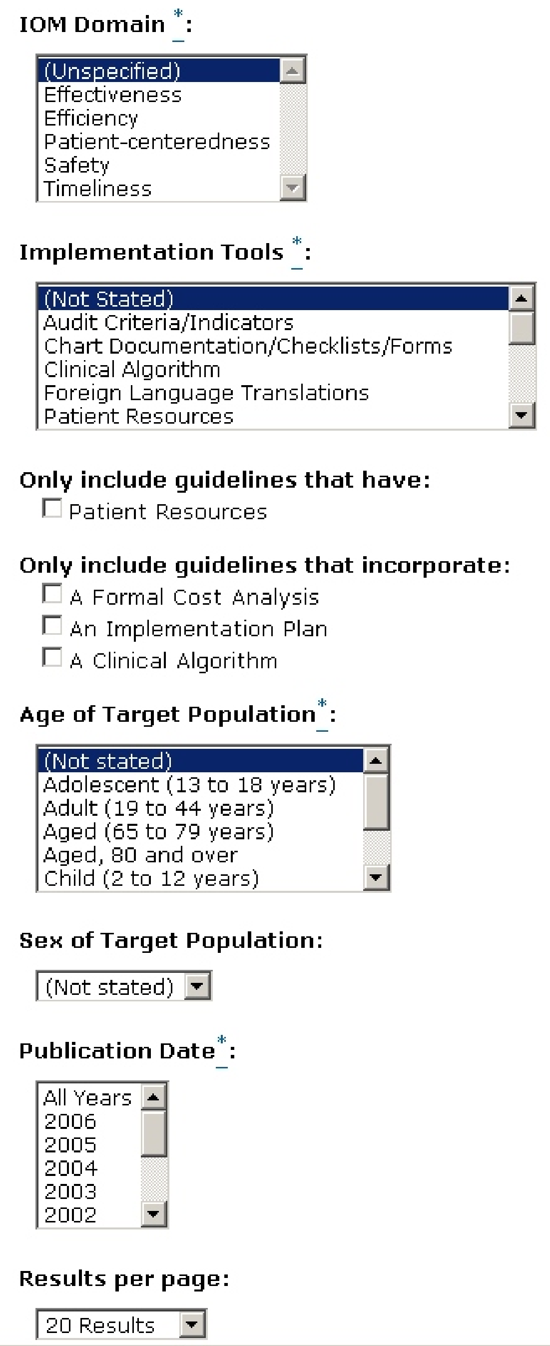

Figure 3: More options for making the query more precise.