Rethinking Advanced Search

Tony Russell-Rose, UXLabs

tgr@uxlabs.co.uk

Abstract

Knowledge workers (such as healthcare information professionals, patent agents and legal researchers) need to create and execute search strategies that are accurate, repeatable and transparent. The traditional solution is to use the proprietary line-by-line "query builders" offered by database vendors. However, these offer limited support for error checking or validation, and their output can often be compromised by errors and inefficiencies. In this article, we explore a new approach to search strategy formulation in which concepts are expressed as objects on a two-dimensional canvas and relationships are articulated using direct manipulation. This eliminates many sources of syntactic error, makes the query semantics more transparent, and offers new ways to validate, share and reproduce search strategies and best practices.

Introduction

Medical knowledge is growing so rapidly that it is difficult for healthcare professionals to keep up. As the volume of published studies increases year by year, the gap between research knowledge and professional practice grows ever wider. Systematic literature reviews can play a key role in closing this gap, by synthesising the complex, incomplete and at times conflicting findings of biomedical research into a form that can readily inform health decision making.

However, undertaking a systematic review is a resource-intensive and time consuming process, sometimes taking years to complete. Even rapid evidence assessments, designed to provide quick summaries of what is known about a topic or intervention, can take as long as three to six months. Moreover, new research findings may be published in the interim, meaning that systematic reviews can be out of date or missing key evidence from the moment they are published.

At its heart, the process of systematic review relies on painstaking and meticulous searching of multiple literature sources. These include published literature sources such as MEDLINE and other specialist databases and "grey literature" (i.e. technical reports and other non-peer reviewed sources). The principal way in which such sources are interrogated is through the use of Boolean queries, which utilise a variety of keywords, operators and ontology terms. Reviewers incrementally build complex queries line by line, sometimes involving hundreds of terms, which are combined to form an overall search strategy. For example, here is a search strategy on the subject of "*Galactomannan detection for invasive aspergillosis in immunocompromised patients*":

1 "Aspergillus"[MeSH]
2 "Aspergillosis"[MeSH]
3 "Pulmonary Aspergillosis"[MeSH]
4 aspergill*[tiab]
5 fungal infection[tw]
6 (invasive[tiab] AND fungal[tiab])
7 1 OR 2 OR 3 OR 4 OR 5 OR 6
8 "Serology"[MeSH]
9 Serology"[MeSH]
10 (serology[tiab] OR serodiagnosis[tiab] OR serologic[tiab]) 11 8 OR 9 OR 10
12 "Immunoassay"[MeSH]
13 (immunoassay[tiab] OR immunoassays[tiab])
14 (immuno assay[tiab] OR immuno assays[tiab])
15 (ELISA[tiab] OR ELISAs[tiab] OR EIA[tiab] OR EIAs[tiab])
16 immunosorbent[tiab]
17 12 OR 13 OR 14 OR 15 OR 16
18 Platelia[tw]
19 "Mannans"[MeSH]
20 galactomannan[tw]
21 18 OR 19 OR 20
22 11 OR 17 OR 21
23 7 AND 22

The choice of search strategy is critical in ensuring that the process is sufficiently exhaustive and that the review is not biased by easily accessible studies. In addition, the strategy needs to be transparent and repeatable, so that others may replicate the methodology. However, there are often mistakes in search strategies reported in the literature that prevent them from being executed in their published form. In one sample of sixty three MEDLINE strategies, at least one error was detected in 90% of these, including spelling errors, truncation errors, logical operator error, incorrect query line references, redundancy without rationale, and more.

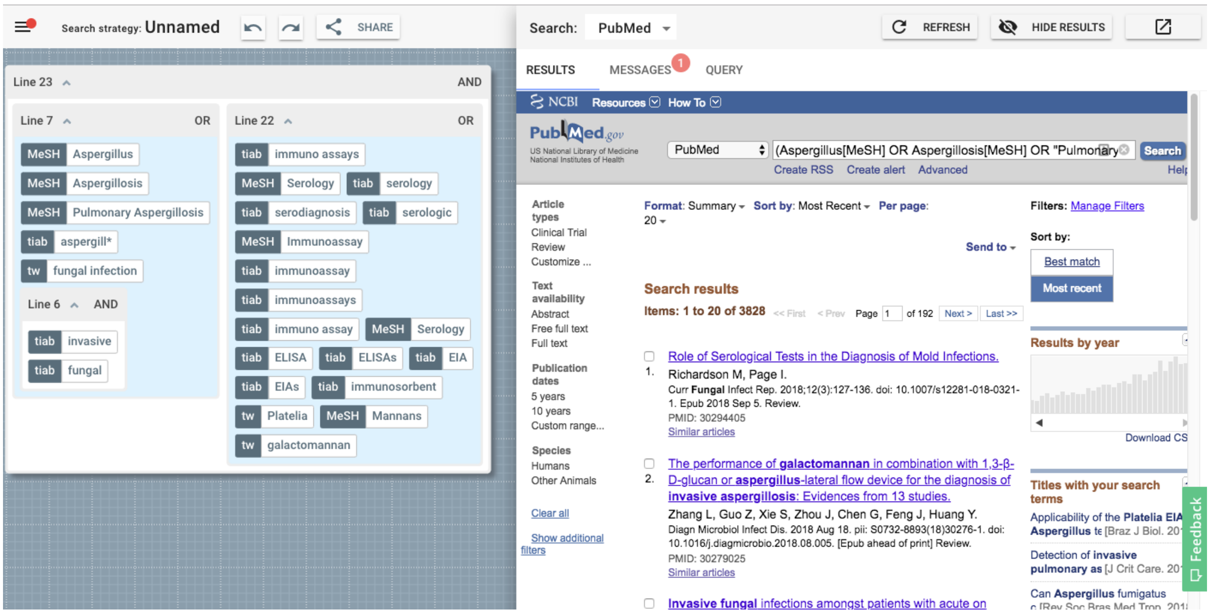
Evidently, despite the painstaking attention to detail of many dedicated individuals, creating effective search strategies is prone to error, often relying on manual processes with limited technological support. Moreover, once completed, strategies are typically published as text-based documents, and are thus rarely directly executable in their native form. This compromises their ability to be reproduced by others and often results in unnecessary duplication.

So, what can be done about this? Well, one approach is to rethink exactly what we mean by “advanced search”. However, if we were designing a framework for structured searching from scratch, the command line paradigm is probably not the ideal place to start. Those line numbers are more a reflection of the days when searches were facilitated by command line instructions to remote databases, and in that respect, they represent the past, not the future. In this article, we explore an alternative approach which we call 2dSearch.

A visual approach to systematic searching

At the heart of 2dSearch is a graphical editor which allows the user to formulate search strategies using a visual framework. Instead of entering Boolean strings into one-dimensional search boxes, queries are formulated by combining objects on a two-dimensional canvas. Concepts can be simple keywords or attribute:value pairs representing controlled vocabulary terms (for example, [MeSH terms]) or database-specific search operators (for example, [field tags] and other commands). They can be combined using Boolean (and other) operators to form higher-level groups and then iteratively nested to create expressions of arbitrary complexity. Groups can be expanded or collapsed on demand to facilitate transparency and readability.

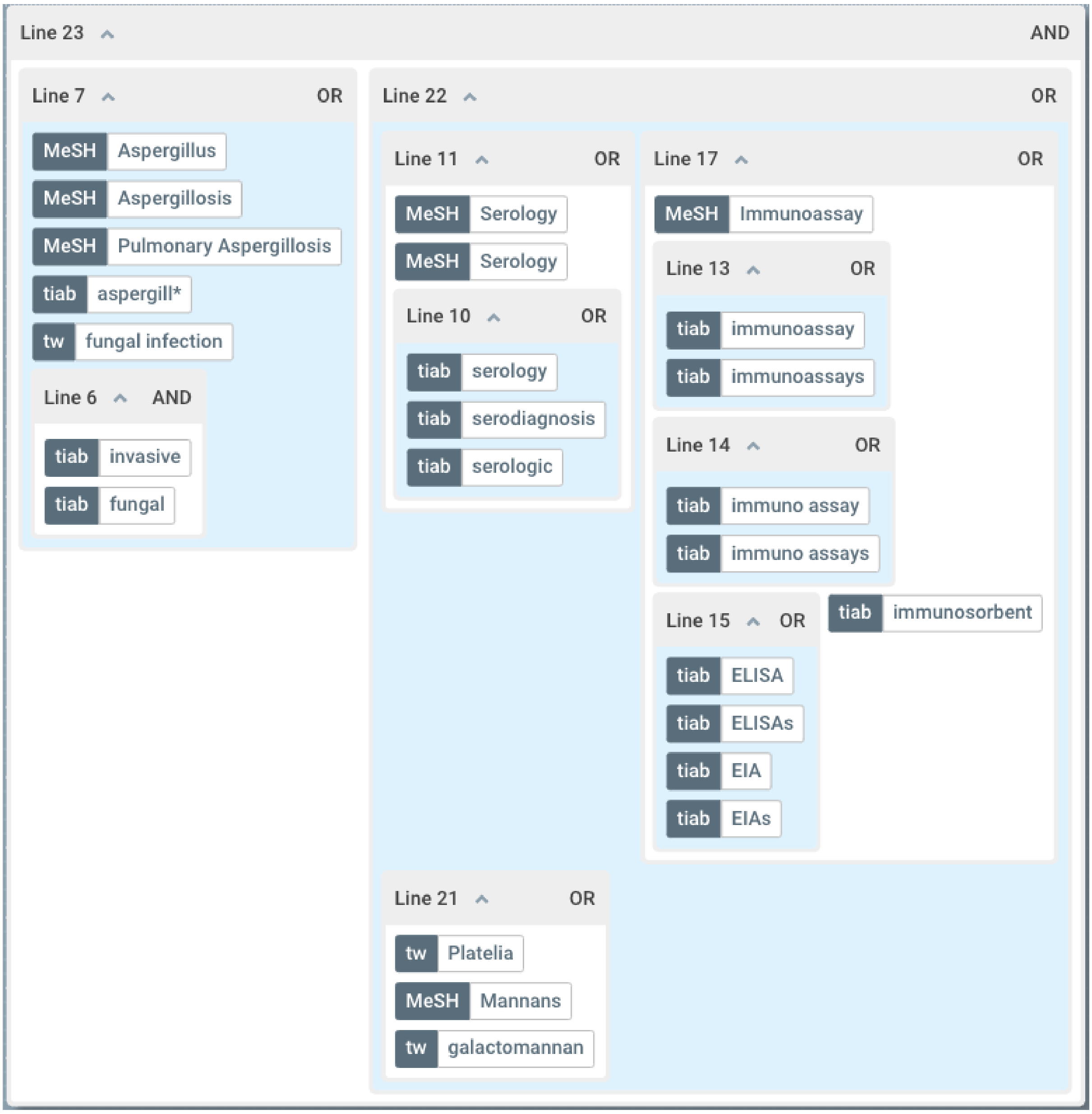
The application itself consists of two panes: a query canvas on the left and a search results pane on the right (which can be resized or detached in a separate tab or window):



The canvas itself can be resized or zoomed and features an “overview” widget which allows the user to view or navigate to elements that may be outside the current viewport. Adopting design cues from Google’s [Material Design language], a sliding menu is offered on the left, providing file I/O and other options. This is complemented by a navigation bar across the top which provides support for common document-level functions such as naming and sharing search strategies.

Although 2dSearch supports the creation of complete strategies from a blank canvas, its function and value are most readily understood by reference to an existing (i.e. text-based) search strategy, such as the example shown above. A trained professional may be able to mentally “parse” the sequence of commands shown and interpret the general approach, but without associated documentation it is difficult to understand exactly what the searcher intended. Moreover, it is difficult to optimise, debug or re-use strategies expressed in this form.

However, when this strategy is opened using 2dSearch, its structure becomes much more apparent:



It can be seen that the overall expression consists of a conjunction of two disjunctions (Lines 7 and 22), the first of which articulates variations on the fungal infection concept, while the latter contains various nested disjunctions to capture the diagnostic test (serology) and associated procedures. Evidently, the line numbers themselves are somewhat arbitrary in this context, having served an original purpose analogous to that of line numbering in 1st generation BASIC. However, by displaying them as nested groups with transparent structure, 2dSearch offers support for abstraction, whereby lower-level details can be hidden and higher-level structure revealed. Moreover, it is now possible to give meaningful names to subgroups, so that they can be saved and reused as modular components.

Although visualisation of search strategies in this manner can offer immediate utility, the true value of the approach is not so much in the *information* design, but in the *interaction* design. For example, to edit the expression, the user can move terms from one block to another using direct manipulation and create new groups simply by combining terms. They can also cut, copy, delete, and lasso multiple objects. If they want to understand the effect of one block in isolation, they can execute it individually. Conversely, if they want to remove one element from consideration, they can temporarily disable it. It is also possible to edit the content inline, interchanging Mesh terms with keywords and field tags as required. In each case, the effects of each editing operation are displayed in real time in the adjacent search results pane.

Validation and reproducibility

It is common for healthcare information professionals to want to search more than one database, particularly when undertaking a systematic literature review. In practice, this requires a process of "translation" of the search strategy to match the syntax of the target database and the search operators it supports. For a relatively simple query this may not be a major undertaking, particularly if such operators form a relatively small proportion of the overall search strategy. However, the user still has to understand which elements are platform-specific, identify the closest equivalent in the other database and manually edit their query, all of which is laborious and time consuming.

2dSearch provides elementary support for search strategy translation in the form of a "Messages" tab on the results pane. This serves a purpose similar to a console or messages pane in a software IDE, alerting the user to compilation issues and offering advice, fixes and workarounds. For example, if the user tries to execute via PubMed a query string containing operators specific to Google Scholar, an alert is shown listing the unknown operators. In due course, this mechanism could be extended to offer a greater degree of interactive support for the translation of strategies across databases. In addition, 2dSearch also offers the potential for search strategy optimisation by identifying redundancy (for example, spurious brackets or duplicate elements) and comparison of canonical representations.

In closing

2dSearch is a framework for search query formulation in which information needs are expressed by manipulating objects on a two-dimensional canvas. Transforming logical structure into physical structure mitigates many of the shortcomings of Boolean strings, eliminates many sources of syntactic error and makes the query semantics more transparent. Librarians and researchers can still write line by line Boolean queries if they wish, but 2dSearch offers new ways for them to be validated, shared and made reproducible.

We are currently working on providing better support for cross-platform integration, which allows a given search strategy to be applied to more than one database. We recognise that providing accurate and reliable translations of search strategies is a significant undertaking, often requiring skilled human judgment. But the point is that it represents something far greater: the prospect of a *universal language for search*, in which information needs can be articulated in a *generic manner*, with the task of mapping to the semantics of an underlying database being delegated to platform-specific adapters. Such a development could have profound implications for the way in which search skills are taught, learnt and applied.

In due course, we hope to undertake a formal, user-centric evaluation, particularly in relation to traditional query builders. In the meantime, try out 2dSearch for yourself, and let us know what you think.

Editor's postscript

Tony Russell-Rose is Director of UXLabs, a research and design consultancy specialising in complex search and information access applications. Previously, Tony was Manager of User Experience at Oracle and editor of the Endeca UI Design Pattern Library, an online resource dedicated to best practice in the design of search and discovery experiences. He holds a PhD in human-computer interaction, an MSc in cognitive psychology and a first degree in engineering, majoring in human factors. He is also Honorary Visiting Fellow at the Centre for Interactive Systems Research, City University, London.

His one-day course on "Search usability: filters and facets" will be held at CILIP's headquarters in London on Thursday 25th April 2019. The course will provide an introduction to the design of search user experiences with a focus on the fundamental principles of faceted search and navigation. For more information go here.