PubMed Assistant: A Biologist-Friendly Interface for Enhanced PubMed Search

Jing Ding

Department of Electrical and Computer Engineering,
Iowa State University, Ames, IA 50011, USA

LaRon M. Hughes

Department of Electrical and Computer Engineering,
Iowa State University, Ames, IA 50011, USA

Daniel Berleant*

Department of Electrical and Computer Engineering,
Iowa State University, Ames, IA 50011, USA

Andy W. Fulmer

The Procter and Gamble Company, Cincinnati, OH 45252, USA

Eve Syrkin Wurtele

Department of Genetics, Development and Cell Biology,
Iowa State University, Ames, IA 50011, USA

Running head: PubMed Assistant

---

* To whom correspondence should be addressed

ABSTRACT

Summary: MEDLINE is one of the most important bibliographical information sources for biologists and medical workers. Its PubMed interface supports Boolean queries, which are potentially expressive and exact. However, PubMed is also designed to support simplicity of use at the expense of query expressiveness and exactness. Many PubMed users have never tried explicit Boolean queries. We developed a Java program, PubMed Assistant, to make literature access easier in several ways. PubMed Assistant provides an interface that efficiently displays information about the citations, and includes useful functions such as keyword highlighting, export to citation managers, clickable links to Google Scholar, and others which are lacking in PubMed.

Availability: PubMed Assistant and a detailed online manual are freely available at http://metnetdb.gdcb.iastate.edu/browser under a GPL (GNU General Public License).

Contact: berleant@iastate.edu, dingjing@iastate.edu

INTRODUCTION

MEDLINE (National Library of Medicine, 2004b) is one of the most important sources of literature citations for biologists, medical workers, and others. Its collection is broad, comprehensive (~15,000,000 citations collected from 4,600 journals dating back to the 1960's), and up-to-date (~10,000 new citations added per week). Its web interface, PubMed (National Library of Medicine, 2004c), is easy to use. A user types in one or more keywords as a query, and MEDLINE returns citations accordingly. However, a citation containing the keywords is not necessarily relevant to the user's interests. To

find the relevant citations, the user has to sift through the retrieved results one by one. This can be tedious. Although the "Limits[1]" function of PubMed, which restricts retrieval to a specified field (i.e. title, title/abstract, MeSH, etc.) and/or a particular time frame can be helpful, Boolean queries provide great additional flexibility and expressiveness to the query system. For example, "*dna [mh] AND crick [au] AND 1993 [dp]*" will find citations on DNA authored by Crick in 1993. Yet the use of explicit Boolean queries is hindered by the relative difficulty of composing them. In fact, many PubMed users have never tried them.

To make Boolean queries user-friendlier and more accessible to PubMed users as well as to add other useful functionalities lacking in PubMed, we developed a Java program, PubMed Assistant. An introduction is provided below. Further details can be found in its online manual[2].

PROGRAM OVERVIEW

PubMed Assistant is a stand-alone Java application that, like PubMed, serves as an interface to MEDLINE. It sends queries to MEDLINE via NCBI Entrez Utilities' ESearch service, retrieves citations via the EFetch service, and finds related articles with the ELink service (National Library of Medicine, 2004a). Unlike PubMed, however, it includes (1) a visual query editor, which eases the editing of Boolean queries; (2) a specialized citation browser, which displays citations more compactly and comprehensively and makes it easier to navigate from one citation to another; (3) an automatic and interactive query refinement tool, AutoQuery; and (4) a collection of

---

[1] http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?CMD=Limits&DB=pubmed
[2] http://metnetdb.gdcb.iastate.edu/browser/manual.htm

handy utility tools which provide quick and easy connections to other frequently used applications. These include exporting to citation managers and one-click Google[3]/Google Scholar[4] searching.

A typical use scenario includes the following steps. The user:

1. searches using a simple query, which returns hundreds or even thousands of hits;

2. sifts through the first few hits, marking some of them as relevant;

3. invokes AutoQuery to formulate a refined query automatically;

4. manually modifies the refined query in the visual query editor if necessary;

5. searches PubMed again with the refined query, resulting in a set of hits that is likely to be considerably smaller and higher in quality;

6. sifts through the improved collection of hits, marking the relevant ones; and

7. outputs the final result to a citation manager.

Visual query editor

The difficulties that hinder wide acceptance of Boolean queries among biologists are summarized in Table 1. Also in the table are the solutions provided by the visual query editor. One notable feature is the tree representation of a query, where Boolean operators are internal nodes and keywords are terminal (leaf) nodes (Figure 1). Operator precedence is represented intuitively as the hierarchical structure of a tree, eliminating the tedious and error-prone process of parenthesis balancing. Interpretation of a query can be done branch by branch from the top down or from the bottom up by expanding or collapsing internal nodes. Restructuring a query is done as a drag-and-drop operation on a tree, which is more naturally suited to the syntactic structure of such queries than cut-

---

[3] http://www.google.com
[4] http://scholar.google.com

and-paste operations in a text editor. The search field options (such as title, title/abstract, MeSH, etc.) are provided in a drop-down list so that users do not have to memorize them.

<div style="border:1px solid">Table 1</div>

<div style="border:1px solid">Figure 1</div>

In addition to the visual editor, PubMed Assistant also has a simple query editor built into the main user interface (Fig. 2). This editor has capabilities similar to PubMed's "Limits" page. It is for occasions when advanced Boolean queries are not necessary.

<div style="border:1px solid">Figure 2</div>

Specialized browser

The following features of the specialized browser are designed to help users identify relevant citations quickly and efficiently (see the citation panel in the lower part of the main interface, Fig. 2).

- One-click (or keystroke) navigation among citations, instead of switching back and forth between the query result page and the abstract pages as in PubMed.

- Compact and comprehensive display of a citation including title, abstract, MeSH list (if available), chemical list (if available), authors, and journal information.

- Keyword highlighting, intended to draw user attention to the most relevant locations in a citation. The highlighter uses a fuzzy (approximate) string-matching algorithm to deal with keyword spelling variations. It also utilizes a local synonym dictionary to identify acronyms and/or other abbreviations.

AutoQuery

All too often, a PubMed search with two or three keywords returns hundreds or even thousands of hits, with only a small proportion relevant to the user's interest. The AutoQuery module is designed to help users quickly and interactively find the most relevant hits.

AutoQuery takes as input a list of relevant citations and optionally the original simple query. It automatically formulates a refined query using the most commonly used words (excluding stopwords) in the list of citations, combined with the original query if included. The stringency (i.e. how many words to formulate the new query with) is a user-adjustable parameter, and helps determine the degree of query refinement (e.g. the reduction in hit set size). Users can interactively adjust the stringency value based on the number of returned hits. The effectiveness of refinement in terms of the relevancy to users' interest is currently under investigation.

Utility tools

A collection of utility tools provides convenient connections to other popular applications. These utilities include the following.

- Export to citation managers. Supported formats include BibTex, EndNote[5], and Ris (Procite[6] and Reference Manager[7]).

- Quick link to PubMed. The PMIDs in the hits table (Fig. 2) are links that will open the original citations in PubMed. In addition, a user can conveniently copy the author names and/or user-selected keywords in the title or the abstract to the query panel (Fig. 2), and perform a PubMed search.

---

[5] http://www.endnote.com/
[6] http://www.procite.com/
[7] http://www.refman.com/

- Quick link to Google/Google Scholar. The "Cited" column in the citation table is reserved for the "Cited by" field in Google Scholar's search result, which shows how many times an article has been cited by other articles according to Google Scholar. Google Scholar is still in beta testing phase as of this writing, so no API is yet available for other applications to extract specific information. The number also provides a clickable link to a search in Google Scholar for the exact article. A user can also perform Google/Google Scholar searches within the main interface by selecting the keywords in the title or the abstract and clicking the corresponding button.

CONCLUSION

We have developed an open source Java application, PubMed Assistant, for enhanced PubMed searches. The enhancements include easier Boolean query editing, iterative query refinement, and more convenient browsing and use of search results.

ACKNOWLEDGEMENTS

REFERENCES

National Library of Medicine (2004a) Entrez Utilities.
http://eutils.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.html

National Library of Medicine (2004b) MEDLINE.
http://www.nlm.nih.gov/pubs/factsheets/medline.html

National Library of Medicine (2004c) PUBMED.
http://www.ncbi.nih.gov/entrez/query.fcgi

Table 1. Difficulties in editing advanced Boolean queries and the solutions provided by the visual query editor.

Difficulties	Solutions

Keeping track of operator precedence: does "a AND b OR c" mean "(a AND b) OR c," or "a AND (b OR c)."	Transform an atomic query unit*:
	    a OP b OP c $\Rightarrow$ OP (a, b, c).
	Normalize a mixed unit (insert omitted parentheses):
	    a OP1 b OP2 c $\Rightarrow$ (a OP1 b) OP2 c $\Rightarrow$ OP2 (OP1 (a, b), c)

Building a mental picture of the query's meaning (as illustrated in Fig. 1). This is error-prone if the query is complex with multiple levels of parentheses.	Represent a transformed and normalized query as a tree with operators as internal nodes and keywords as leaf nodes (Fig. 1). Collapsing and expanding internal nodes enables building mental pictures branch by branch from the top down or from the bottom up.
Balancing parentheses. While it takes little effort to balance "a AND (b OR c)," it is more tedious to check the correctness of "(((a AND b) OR c) NOT (e OR (f NOT g))) OR g."	Parentheses are implied in the tree structure. There are no explicit parentheses in a fully expanded tree at all, so there is nothing to balance.
Rearranging a query's structure is difficult, because a user has to re-balance parentheses.	Rearranging a query is equivalent to dragging-and-dropping a branch from its original parent node to a new one. There are no parentheses to re-balance.
Memorizing or looking up the abbreviations of search fields.	Select search fields from a list, instead of typing them in by hand.

* A query unit is defined as the Boolean expression within a pair of balanced parentheses. An atomic unit is a query unit that has only one type of Boolean operator, while a mixed unit contains two or three types of operators.
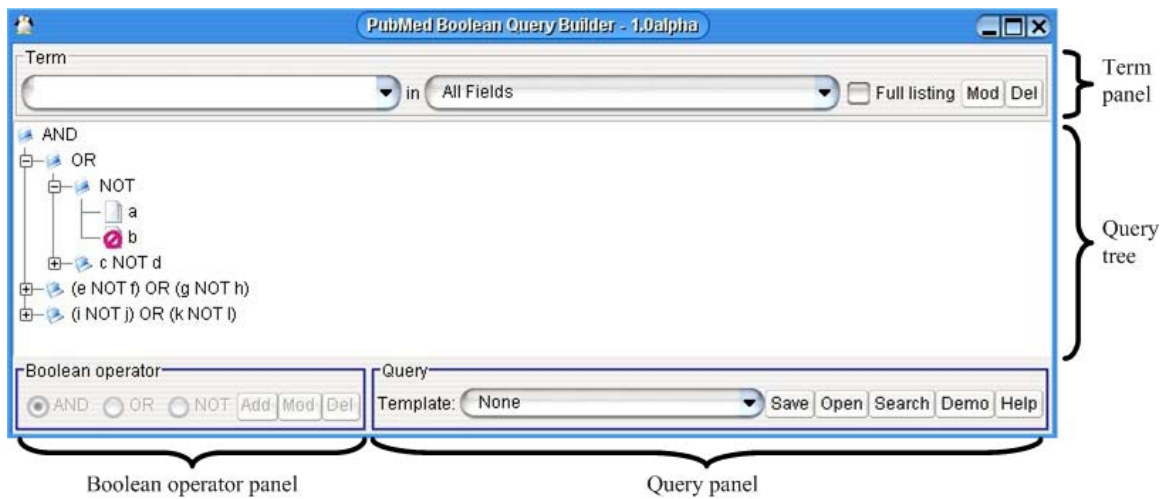
Figure 1. The visual query editor showing the tree representation of a query. It represents the following query:
((a NOT b) OR (c NOT d)) AND ((e NOT f) OR (g NOT h)) AND ((i NOT j) OR (k NOT l)).
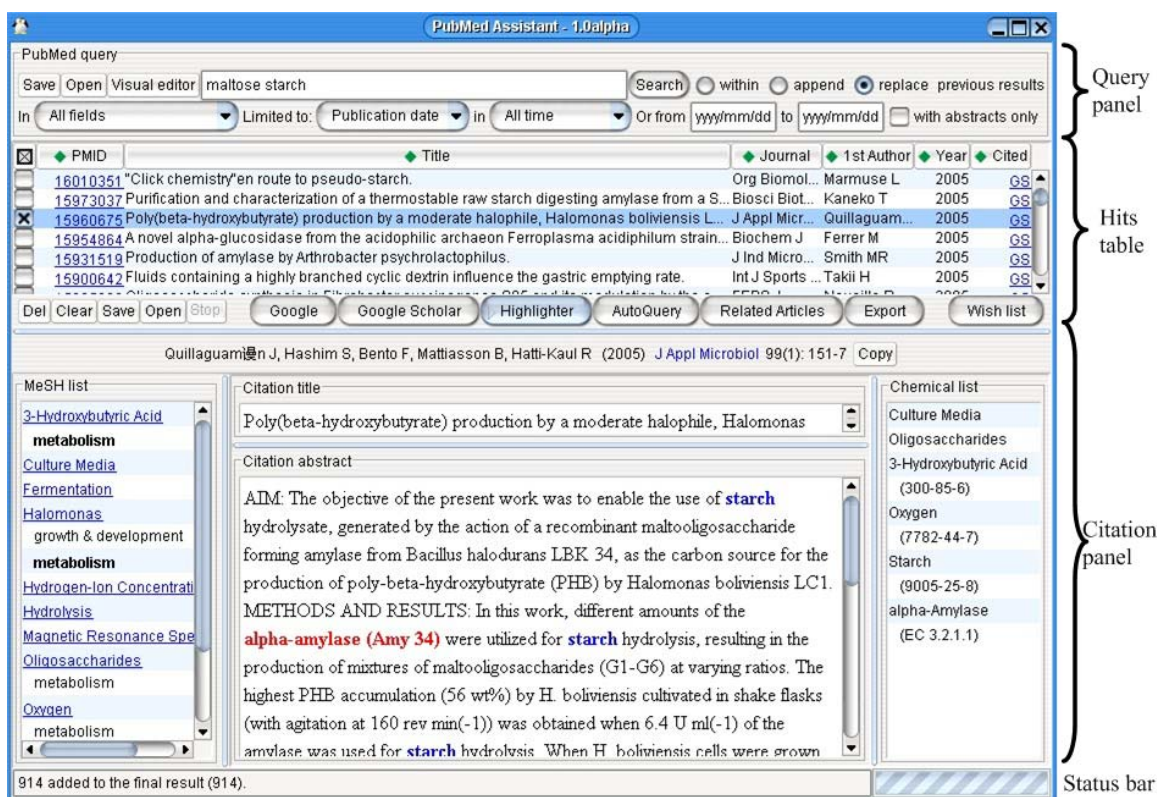
Figure 2. The main user interface of PubMed Assistant. Functionalities are organized in three sections. The top section (query panel) is a simple query editor, which has capabilities similar to PubMed's "Limits" page. A visual query editor can be opened from here to edit advanced Boolean queries. The midsection (hits table) is where a user manages the returned citations, e.g. marks relevant hits, saves the hits to or loads them from local hard disks, exports to citation managers, links to PubMed, Google and Google Scholar, etc. The bottom section (citation panel) is a specialized browser that displays a citation's title, abstract, MeSH terms, chemical list, and authors. Note that some words in the abstract are highlighted, showing matches to the query terms (blue) or user specified keywords (red).