Exploring Biomedical Databases with BioNav

Abhijith Kashyap
CSE Dept
SUNY Buffalo
Buffalo, NY 14260
rk39@cse.buffalo.edu

Vagelis Hristidis
School of Computing and Information Sciences
Florida International University
Miami, FL 33199
vagelis@cis.fiu.edu

Michalis Petropoulos
CSE Dept
SUNY Buffalo
Buffalo, NY 14260
mpetropo@cse.buffalo.edu

Sotiria Tavoulari
Pharmacology Dept
Yale University
New Haven, CT 06520
sotiria.tavoulari@yale.edu

1. INTRODUCTION
The last decade has been marked by unprecedented growth in both the production of biomedical data and the amount of published literature discussing it. The MEDLINE database, on which the PubMed search engine operates, contains over 18 million citations, and the database is growing at the rate of 500,000 new citations each year [7]. Keyword search queries on these databases return a large results set from which only a small portion is relevant for the user. Many solutions have been proposed to address this problem—commonly referred to as information-overload [2,3]. These approaches can be broadly classified into two classes: ranking and categorization, which can also be combined.

Figure 1. Static Navigation on the MeSH Concept Hierarchy

BioNav belongs primarily to the categorization class, which is ideal for this domain given the rich concept hierarchies available for biomedical data, such as MeSH [5]. Each citation in MEDLINE is associated with several MeSH concepts in two ways: (i) by being explicitly annotated with them, and (ii) by mentioning them in their text. Since these associations are provided by PubMed, a relatively straightforward interface to navigate the query result would first attach the citations to the corresponding MeSH concept nodes and then let the user navigate the concept hierarchy. Figure 1 displays a snapshot of such an interface where shown next to each node label is the count of distinct citations in the subtree rooted at that node. For this example, we assume that the user queries MEDLINE for the nucleoprotein “prothymosin” and his personal interests are reflected in the two indicated concepts, corresponding to two independent lines of research related to prothymosin. A typical navigation starts with revealing the children of the root ranked by their citation count, and is continued with expanding one or more of them, revealing their ranked children and so on. Further, the user may click on a concept and inspect the attached citations. A similar interface and navigation method is used by GoPubMed [6] and e-commerce sites, such as Amazon and eBay.

The above static—same for every query result—navigation method is problematic when the MeSH hierarchy (or one with similar properties) is used for categorization for the following reasons:

• The massive size of the MeSH hierarchy (over 48,000 concept nodes) makes it challenging for the users to effectively navigate to the desired concepts and browse the associated records.

• A substantial number of duplicate citations are introduced in the navigation tree of Figure 1, since each one of the 313 distinct citations is associated with several concepts. Specifically, the total count of citations in Figure 1 is 40,195.

BioNav, first proposed in [1], introduces a dynamic navigation method that depends on the particular query result at hand. The query results are attached to the corresponding MeSH concept nodes as in Figure 1, but then the navigation proceeds differently. The key action on the interface is the expansion of a node that selectively reveals a ranked list of descendant (not necessarily children) concepts, instead of simply showing all its children.

Figure 2. BioNav Interface after Querying for “prothymosin” and its Associated Subtree Information Window
**BioNav Interface.** Figure 2 shows the state of the BioNav interface after querying for “prothymosin”. The root of the MeSH tree can be seen on the left pane. The right pane shows the results under the current node of the navigation tree of the left pane. The user can also view more information about a subtree rooted at a given concept node by clicking on the icon that appears next to each concept label. The table of the pop-up window in Figure 2 shows various characteristics of the current subtree, including the fact that the 313 citations in the query result are spread over 3940 concept nodes.

![Image](image_url)

(a)

- **MeshUIX**
  - Amino Acids, Peptides, and Proteins (319) >>>
  - Transcription Factors (46) >>>
  - Nucleotides (14) >>>
  - Chromatin (11) >>>
  - Histones (9) >>>

(b)

- **MeshUIX**
  - Amino Acids, Peptides, and Proteins (319) >>>
  - Blood Proteins (32) >>>
  - Blood Proteins (32) >>>
  - Blood Proteins (32) >>>

(c)

- **MeshUIX**
  - Amino Acids, Peptides, and Proteins (319) >>>
  - Transcription Factors (46) >>>
  - Nucleotides (14) >>>

(d)

**Figure 3. BioNav Navigations**

**BioNav Navigation.** Figure 3a shows the initial expansion of the root node where only 8 (highlighted) descendants are revealed compared to 98 children shown in Figure 1. The concepts are ranked by their relevance to the user query and the number of them revealed depends on the characteristics of the query results. Next, assuming the user is interested in the “Amino Acids...” node and judging that the 310 attached citations is still a big number, she expands it by clicking on the “&gt;&gt;&gt;” hyperlink next to it in Figure 3b. The user inspects the 6 concepts revealed and decides that she is not interested in any of them. Hence, she expands the “Amino Acids...” node one more time in Figure 3c, revealing 4 additional concepts. Note that “Nucleoproteins” is an example of a descendant node being revealed, since its parent node “Proteins” (shown in Figure 1) is not revealed in Figure 3c. In Figure 3d, the user expands the “Nucleoproteins” node and reveals “Histones”, one of the two key concepts for the query. Note that to reach “Histones” using the BioNav navigation method only 23 concepts are revealed, after 4 node expansions, compared to 152 concepts, also after 4 expansions, with the static navigation method of Figure 1.

**2. SYSTEM OVERVIEW**

The MeSH concept hierarchy is a labeled tree [5], where the label of a child concept node is more specific than the one of its parent. Once the user issues a keyword query, PubMed--BioNav uses the Entrez Programming Utilities (eUtils) [4]--returns a list of citations, each associated with several MeSH concepts. BioNav constructs a Navigation Tree by attaching to each concept node of the MeSH concept hierarchy a list of its associated citations and removing all nodes with no citations, while preserving the ancestor-descendant relationship. The Navigation Tree $T(V,E,r)$ is the maximum embedding of an initial navigation tree $T(N,E,r)$ such that no node $n \in V$ is labeled with an empty results list $L(n)$, excluding the root (in order to maintain the tree structure and avoid the creation of a forest).

![Image](image_url)

**Figure 4. Navigation Tree, EdgeCut and Component Subtrees**

We model a node expansion at a given navigation step as an EdgeCut in the navigation tree. In Figure 4, the dashed line illustrates the EdgeCut corresponding to the expansion of the node “Amino Acids...”. This expansion reveals the highlighted concepts of Figure 4, which include a subset of the highlighted concepts in Figure 3c. The EdgeCut consists of the edges (“Proteins”, “Transcription Factors”) and (“Proteins”, “Nucleoproteins”). A valid EdgeCut of a tree $T(V,E,r)$ is an EdgeCut $C \subseteq E$ such that no two edges in $C$ appear in a path from the root to a leaf node. We only consider valid EdgeCuts, because invalid EdgeCuts lead to unintuitive navigations.

**Component Subtrees** An EdgeCut causes the creation of two types of component subtrees, a single upper and possibly multiple lower. Figure 4 shows two lower component subtrees, rooted at “Transcription Factors” and “Nucleoproteins”, and an upper component subtree comprising of the node being expanded “Amino Acids...” and all nodes not in any of the lower component subtrees.

**2.1 Navigation and Cost Model**

BioNav initiates a navigation by constructing the initial results tree and displaying its root to the user. Subsequently, the user...
navigates the tree by performing one of the following actions on a given component subtree $I(n)$ rooted at concept node $n$:

1. **EXPAND $I(n)$**: The user clicks on the ">>>" hyperlink next to node $n$ and causes an $\text{EdgeCut}(I(n))$ operation to be performed on it, thus revealing a new set of concept nodes from the set $I(n)$.

2. **SHOWRESULTS $I(n)$**: By performing this action, the user sees the results list $L(I(n))$ of citations attached to the component subtree $I(n)$.

3. **IGNORE $I(n)$**: The user examines the label of concept node $n$, ignores it as unimportant and moves on to the next revealed concept.

This navigation process continues until the user finds all the citations she is interested in. The cost of a navigation is computed as follows: We assign (i) cost of 1 to each newly revealed concept node that the user examines after an EXPAND action, (ii) a cost of $B$ (determined empirically) to each EXPAND action the user executes, and (iii) cost of 1 to each citation displayed after a SHOWRESULTS action. BioNav estimates the navigation cost by taking in to account the probability that the user will execute an EXPLORE or SHOWRESULTS action at each step of the navigation. The EXPLORE probability is proportional to the number of unique results in the corresponding component subtree, whereas normalized entropy of the component subtree is used as the SHOWRESULTS probability.

The off-line components populate the BioNav database. The off-line components populate the BioNav database, so that BioNav can be used in a backup scenario.

### On-Line Operation

Upon receiving a keyword query from the user, BioNav executes the same query against the MEDLINE database and retrieves only the IDs (PubMed Identifiers) of the citations in the query result using the ESearch utility [4]. Next, the navigation tree is constructed by retrieving the MeSH concepts associated with each citation in the query result from the BioNav database.

Initially, the root of this navigation tree is shown to the user. Subsequently, when she requests an EXPAND action on the root, the Navigation Subsystem executes a heuristic algorithm to compute the best EdgeCut and the roots of the resulting component subtrees are visualized on the web-interface.

### 3. DEMONSTRATION PLAN

In this demonstration we will show the effectiveness of BioNav in reducing the overall navigation cost as compared to static navigation. Specifically, we show that BioNav, (a) shows significantly fewer number of concept labels as compared to static navigation, (b) selects relevant descendents, as opposed to children, to be revealed at each expansion step and (c) has EXPAND actions comparable to that of static navigation. The BioNav system is available at [http://db.cse.buffalo.edu/bionav](http://db.cse.buffalo.edu/bionav) and the static navigation is available through the information pop up window (right side of Figure 2) for the users to compare in parallel. Further, we provide a set of sample queries at BioNav website for users with no biomedical background. A limitation of the current deployment of BioNav is that we do not allow queries that return more than 3000 results as this would slow down the system. The user is asked to refine such a query before BioNav processes it.

### Backup Scenario

Since BioNav depends on eUtils [4] to download the list of citation IDs and high-level information for each citation returned by a query, we propose to download this data for a set of sample queries.

### REFERENCES


---

**Figure 5. BioNav System Architecture**

#### 2.2 System Architecture

The BioNav system architecture is shown in Figure 5 and consists of two parts. The off-line components populate the BioNav database with the MeSH concept hierarchy and the associations of the MEDLINE citations with MeSH concepts to decrease the on-line response time. The on-line components support BioNav’s web interface and the EXPAND/SOWRESULTS user actions.

#### Off-Line Pre-Processing

The BioNav database is first populated with the MeSH hierarchy (48441 concepts) available at [5]. Next, the associations of MEDLINE citations and MeSH concepts are populated by issuing a query on PubMed for each concept $c$. For each citation $r_i$ returned by the query, we add the association $<c, r_i>$ in our database.