Volume 3, No. 8, August 2012
Journal of Global Research in Computer Science
ISSN-2229-371X
RESEARCH PAPER
Available Online at www.jgrcs.info

# AN EFFECTIVE OPTIMIZATION OF QUERY RESULTS USING MIDDLEWARE LAYERS BASED ON CONCEPT OF HIERARCHIES IN BIOMEDICAL DATABASES

Madduri Venkateswarlu[1] and V. Srinivas[2]

[1]Student, M.Tech (CSE), Anadapuram , Al-Ameer College of Engineering and Technology, VIZAG, A.P, India
[1]maddurivenkateswarlu169@gmail.com
[2]Asst.Professor, (CSE), Anadapuram , Al-Ameer College of Engineering and Technology, VIZAG, A.P, India

***Abstract-*** Search queries on biomedical databases, such as Pub Med, often return a large number of results, only a small subset of which is relevant to the user. Ranking and categorization, which can alsobe combined, have been proposed to alleviate this information overload problem. Result optimization and results categorizationfor biomedical databases are the focus of this work. A natural way to organize biomedical citations is according to their Mesh annotations. MeSH is a comprehensive concept hierarchy used by Pub Med. In this paper, we present the BioIntelR (BIR) system, adopts the BioNav system enables the user to navigate large number of query results by organizing them using the Mesh concept hierarchy. First, BioIntelR (BIR) system prompts the user for the search criteria and the system automatically connects to a middle layer created at the application level which directs the query to the proper valid query path to select correct criteria of the search result from the biomedical database. The query results are organized into navigation tree. At each node expansion step, BIR system reveals only a small subset of the concept nodes, selected such that the expected user navigation cost is minimized. In contrast, to the previous systems, the BIR system outperforms and optimizes the query result time and minimizes query result set for easy user navigation, Data Warehousing.

***Keywords:*** Interactive Data Exploration and Discovery, Search Process, Graphical User Interfaces, Interaction Styles.

## 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[1] database, on which the Pub Med[2,3] search engine operates, contains over 18 million citations and is currently growing at the rate of 500,000 new citations each year [4]. Other biological sources, such as Entrees Gene [5] and OMIM [6], witness similar growth. Biologists, chemists, medical and health scientists are used to searching their domain literature—such as Pub Med—using a keyword search interface. Currently, in an exploratory scenario where the user tries to find citations relevant to her line of research and hence not known apriori, she submits an initially broad keyword-based query that typically returns a large number of results. Subsequently, the user iteratively refines the query, if she has an idea of how to, by adding more keywords, and resubmits it, until a relatively small number of results are returned. This refinement process is problematic because after a number of iterations, the user is not aware if she has over specified the query, in which case relevant citations might be excluded from the final query result. As an example [7].

The BIR system is developed to facilitate the keyword search Unplumbed to using MeSH concept hierarchy. The Proposed system, accepts the user search criteria and specifies the valid query path by using middle layer (shown in fig. 2) constructed at

The application level before running on the Bio Medical databases by incorporating the BioNav methods to get the meaningful results which leads to the further data analysis.

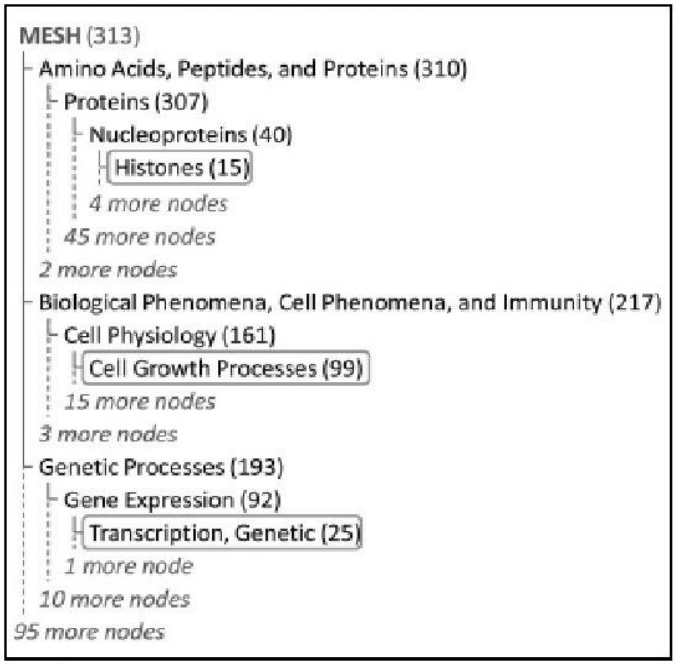



Figure 1: Static Navigation on the MeSH Concept Hierarchy

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* [8,9].
These approaches can be broadly classified into two classes: ranking and categorization, which can also be combined.

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[10].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 Pub Med, 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.

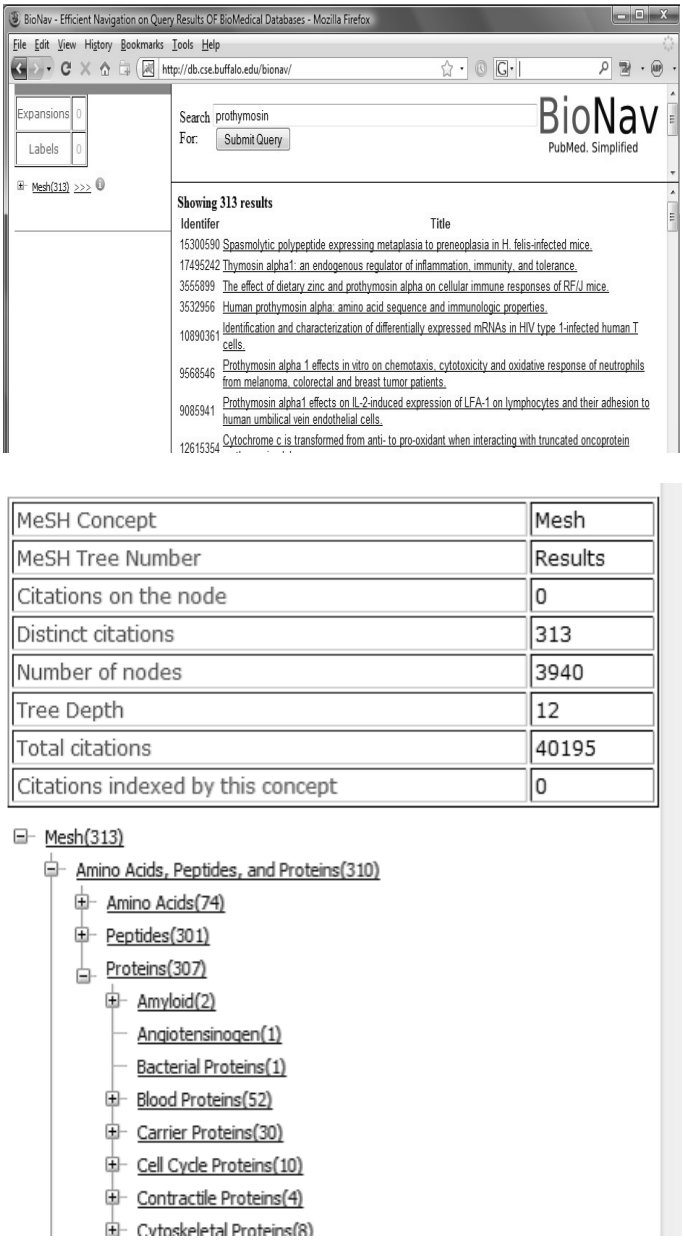
Figure 1 displays a snapshot of such an interface where shown next to each node label is the count of distinct citations in the sub tree rooted at that node. For this example, we assume that the user queries MEDLINE for the nucleoprotein "prothymosin" and his personal interests are reflected in the two 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 aconcept and inspect the attached citations. A similar interface and navigation method is used by Go Pub Med [11] and e-commerce sites, such as Amazon and eBay.

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

a. The massive size of the MeSH hierarchy (with 48,441concept nodes) makes it challenging for the users to effectively navigate to the desired concepts and browse the associated citations.
b. A substantial number of *duplicate* citations are introduced in the navigation tree of Figure 1, since each one of the 313d*istinct c*itations is associated with several concepts. Specifically, the total count of citations in Figure 1 is 40,195.

BioNav, first proposed in [12], 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 shows the state of the Bio Navinterface after querying for "prothymosin". The root of the MeSHtree can be seen on the left pane. The right pane shows the result sunder the current node of the navigation tree of the left pane. The user can also view more information about a sub tree rooted at a given concept node by clicking on the icons that appear next to each concept label. The table of the pop-up window in Figure 2shows various characteristics of the current sub tree, including the fact that the 313 citations in the query result are spread over 3940concept nodes.




| MeSH Concept | Mesh |
|---|---|
| MeSH Tree Number | Results |
| Citations on the node | 0 |
| Distinct citations | 313 |
| Number of nodes | 3940 |
| Tree Depth | 12 |
| Total citations | 40195 |
| Citations indexed by this concept | 0 |

- Mesh(313)
  - Amino Acids, Peptides, and Proteins(310)
    - Amino Acids(74)
    - Peptides(301)
    - Proteins(307)
      - Amyloid(2)
      - Angiotensinogen(1)
      - Bacterial Proteins(1)
      - Blood Proteins(52)
      - Carrier Proteins(30)
      - Cell Cycle Proteins(10)
      - Contractile Proteins(4)
      - Cytoskeletal Proteins(8)

Figure 2: BioNav Interface after Querying for "prothymosin" And it's Associated Sub tree Information Window

This paper is organized as follows: Section II describes related work, Section III describes the Proposed System architecture Work nature, Section IV presents experiment Evolution. Section V presents conclusion and future scope.

## RELATED WORK

Several systems have been developed to facilitate keyword search on Pub Med using the MeSH concept hierarchy. Pub Med itself allows the user to search for citations based on MeSH annotations. A keyword query "his tones [MeSH Terms]" will retrieve all citations annotated with the MeSH term "his tones" in the MeSHhierarchy. The user can also limit her search to a MeSH term by using additional filters, e.g.,"[majr]" to filter out All citations in the query result that don't have the term as their major term. These filters can be

combined by using the Boolean connectives AND, OR, and NOT. This interface poses significant challenges, even to experienced users, since the annotation process is manual and thus prone to errors. The closest to BioNav is Go Pub Med [13],[14], which implements a static navigation method on the results of Pub Med. Go Pub Med lists a predefined list of high-level MeSHconcepts, such as "Chemicals and Drugs," "Biological Sciences,"and so on, and for each one of them displays the top-10 concepts. After a node expansion, its children are revealed and ranked by the number of their attached citations, whereas BioNav reveals selective and dynamic list of descendant (not always children)nodes ranked by their estimated relevance to the user's query. Further, BioNav uses a cost model to decide which concepts to display at each step.

BioNav belongs primarily to the categorization class, which is especially suitable for this domain given the rich concept hierarchies (e.g., MeSH [15]) available for biomedical data. We augment our categorization techniques with simple ranking techniques. BioNavorganizes the query results into a dynamic hierarchy, the navigation tree. Each concept (node) of the hierarchy has a descriptive label. The user then navigates this tree structure, in a top-down fashion, exploring the concepts of interest while ignoring the rest Node that the user is not aware that the relevant results are available specifically on these nodes—she is only interested in narrowing down the results, using a familiar concept hierarchy, instead of examining all the results The above static [7]—same for every query result—navigation method is problematic when the MeSHhierarchy (or one with similar properties) is used for categorization for the following reasons [7].

BioNav introduces a dynamic navigation method that depends on the particular query result at hand and is demonstrated in Fig. 3.The query results are attached to the corresponding MeSH concept nodes as in Fig. 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. Fig. 3a, for example, shows the initial expansion of the root node where only eight (highlighted) descendants are revealed compared to 98 children shown in Fig. 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 ">>>" hyperlink next to it in Fig. 3b. The user inspects the six concepts revealed and decides that she is not interested in any of them. Hence, she expands the "Amino Acids ..."node one more time in Fig. 3c, revealing four additional concepts. Note that "Nucleoproteins" is an example of a descendant node being revealed, since its parent node "Proteins" is not revealed in Fig. 3c. In Fig. 3d, the user expands the "Nucleoproteins" node and reveals "His tones," one of the three key concepts for the query. In the last step of the interaction, the user clicks on the "Histones"hyperlink and the 15 corresponding citations are displayed in a separate frame as shown in Fig. 4. To reach "His tones" using the BioNav navigate

Method, only 23 concepts are revealed, after four node expansions, compared to 152 concepts, also after four expansions, with the static navigation method of Fig. 1. For each expansion, the displayed descendant concepts are chosen in a way that the expected navigation cost is minimized, based on an intuitive navigation cost model. The cost modelestima tes the exploration probability for a node based on its selectivity, that is, the ratio of attached citations before and after the query.

The navigation cost for a concept node is also proportional to the density of the navigation sub tree rooted at this node in terms of citation count.

Intuitively, the selection is done such that every expansionreducesmaximally the expected remaining navigation cost. For example, the reason that "Proteins" is not displayed in fig. 3, is that it is too general given the query results and the original distribution of citations in the PubMed database (details in Sections 3 and 4),and hence displaying it would lead to an expected increase in the user navigation cost, based on the user navigation cost model.In addition to the static hierarchy navigation works mentione d above, there are works on dynamic categorization of query results (e.g., the Crusty search engine [16], or [17-18]), which creation supervised query-dependent results clusters, but do not study how the Clusters should be navigated. BioNav is distinct since it offers dynamic navigation on a predefined hierarchy, as is the MeSH concept hierarchy. Another difference is that BioNav usesa navigation cost model to minimize the navigation cost. We make the following contributions:

a. A comprehensive framework for navigating large query results from PubMed using MeSH, an extensive concept hierarchy used for indexing citations in MEDLINE
b. A formal cost model for measuring the navigation cost in curredby the user.
c. A complexity result proving that expanding the tree in a way that minimizes the user's navigation cost is an NP-Complete problem.
d. An efficient heuristic and a feasible optimal algorithm for minimizing the navigation cost.
e. Experimental results validating the effectiveness of the BioNav system when compared to state-of-the art categorization systems.
f. An online version of the BioNav system is available at http://db.cse.buffalo.edu/bionav.Although we specifically target the biomedical domain in this work; the approach can be directly applied to data sets where tuples are classified using terms from a concept hierarchy. The core of the first contribution has been presented in our preliminary short paper [19].

The MeSH concept hierarchy is a labeled tree [10], where the label of a child concept node is more specific than the one of its parent. Once the user issues a keyword query,

PubMedBioNav uses the Entrees Programming Utilities (etuis) [20].

It returns a list of citations, each associated with several MeSH concepts. BioNavconstructs 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-descendent relationship. The *navigation tree* $T(V,E,r)$is the maximum embedding of an initial navigation tree $T_I(V_I,E_I,r)$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).

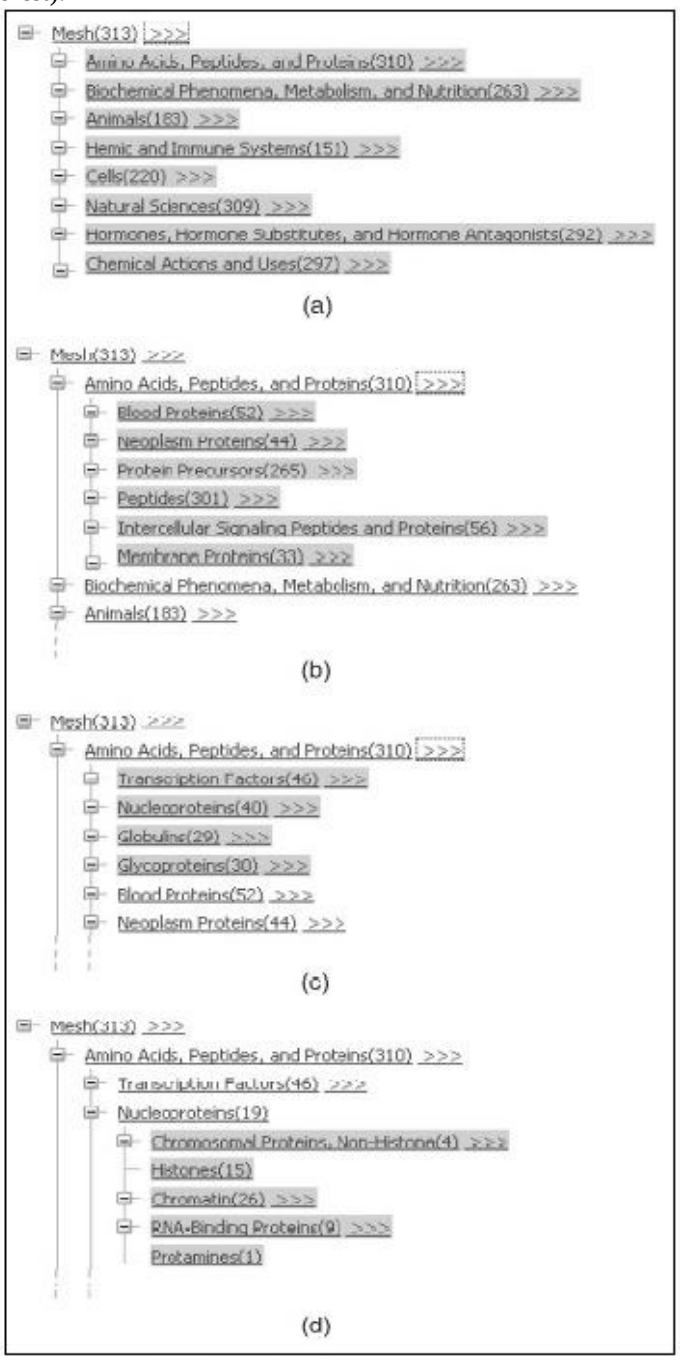



Figure 3: BioNav Navigations

## FRAMEWORK AND BIOINTELR OVERVIEW

Information overload is a common phenomenon encountered by users searching biomedical databases such as PubMed. We encounter this problem; we resolve this problem by optimizing the query result time and minimize query result set for easy user navigation.

### *Architecture of Bio Intel R System:*

The propose BIR system consists combination of:

a. Web interfaces
b. Middle layer
c. Navigation system,
d. Programming utilizes
e. Data base

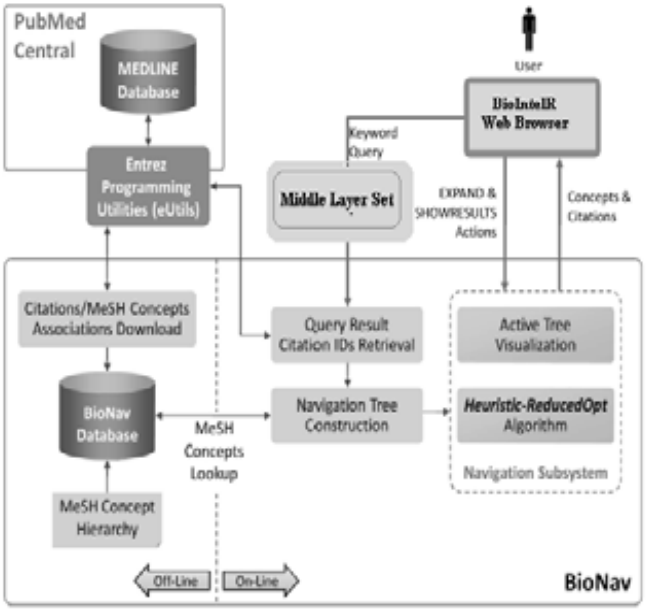



Figure. 4: BioIntelR System Architecture

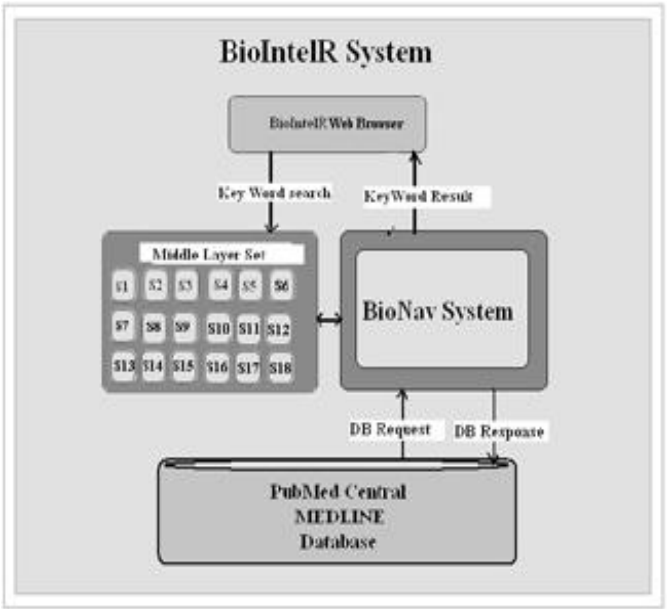



Figure. 5: Bio IntelR System

Upon receiving a keyword query from the user, BioIntelR sends the query and visualize the query results; BioNav executes the same query against the MEDLINE database and retrieves only the IDs (PubMed Identifiers) of the citations in the query result. The user interacts with system by using BioIntelR web browser to find the effective results of the search criteria from PubMed. Previously the BioNav system, once the user issues a keyword query, PubMed—BioNav uses the Entrees Programmed Utilities (eUtils) [21]—return a list of citations, each associated with several MeSH concepts. BioNav constructs an Initial Navigation Tree by attaching to each concept node of the MeSH concept hierarchy a list of its associated citations BioNav reduces the size of the initial navigation tree by removing the nodes with empty results lists, while preserving the ancestor/descendant relationships. The MeSH concept hierarchy is the starting point of the framework and is defined as follows: Concept Hierarchy, Navigation Tree, Valid Edge Cut, Active Tree, Active Tree Visualization [7], and The navigation model of BioNav which is used to device and evaluate algorithms [7].

***Web Interface:***

Web interface is the user interface interacts with the BioIntelRsystem, by specifying the search criteria by specifying the searchkey words to visualize the optimized results from the system.

***Middle Layer:***

The role of the middle is to provide an easy to use and understand interface for user to search criteria against database to get the minimal result set for easy navigation and it reduces search result time. The middle layer is a file mainly consists of the schema of objects is created according underlying database, the file contains connection parameter to connect the database, the middle layer Maps the search keywords to the database and validated path for the search criteria .The layer acts as bridge between the user interface and the database. The schemas that we created must be relevant to the end user business environment and vocabulary.

***Navigation System:***

After the user issues a keyword query, BioNav initiates navigation by constructing the initial active tree (which has a single component tree rooted at the MeSH root) and displaying its root to the user. Subsequently, the user navigates the tree by performing one of the following actions on a given component sub tree rooted at concept node n: EXPAND, SHOWRESULTS, IGNORE, BACKTRACK this navigation process continues until the user finds all the citations she is interested in.

***Programming Utilities:***

Entrees Programming Utilities (eUtils) [21]—returns a list of citations, each associated with several MeSH concepts.

***Data Bases:***

MEDLINE database, fig.4 on which the PubMed search engine operates, contains over 18 million citations and is at a time Growing at the rate of 500,000 new citations each year. The BioNavdatabase is first populated with the MeSH hierarchy, which is available online [15] and has more than 48,000 concept nodes. Then, the BioNav database is populated with the associations of the MEDLINE citations to MeSH concepts.

***Work Nature of the Proposed System:***

The main aim of the proposed system is to search effective results from millions citations, The proposed system BioIntelR (BIR)(contains set of Middle layers and any Bio medical search tool we consider the BioNav System) accepts the user search key words and prompts the user for specific filter fields, the system accepts user request, and choose the effective middle layer from the middle layer set to effectively process request. The layer acts as a bridge between the user interface and BioIntelR system. When a user queries the database fig.4, the middle layers specifies the correct path to run over the large central database.

The proposed system adopts the BioNav system, one the path Is specified and the BioNav system, The BioNav system architecture is shown in Fig. consists of two parts. The offline components populate the BioNav database with the MeSH concept hierarchy and the associations of the MEDLINE citations with MeSH concepts, while the online components support BioNav's web user own interface and the EXPAND-SHOWRESULTS actions of the user.

***Offline Preprocessing:***

The BioNav database is first populated with the MeSHhierarchy, which is available online [15] and has more than 48,000 concept nodes. Then, the BioNav database is populated with the associations of the MEDLINE citations to MeSH concepts. These associations are not directly provided by the Entrees Programming Utilities (eUtils), so we had to implement the following method to infer these associations. For each concept in the MeSH hierarchy, we issued a query on PubMed using the concept as the keyword. For each citation ID in the query result, we added to a table in the BioNav database the tuple <concept; citation ID>. Alternatively, we could determine the associations by using the MeSH concepts that each citation is annotated within the MEDLINE database. This information is available through eUtils. In this case though, the navigation trees of BioNav would not be very informative, since each citation is annotated with 20 concepts on average in MEDLINE, while the PubMed indexing associates each citation with approximately 90 concepts on average (and include the 20from MEDLINE.) Given the number of concepts in the MeSHhierarchy, the number of citations in MEDLINE (_18 million), and the PubMed eUtils restrictions on the number of queries that can be executed within a certain period of time, it took almost 20 days to collect all the <concept; citation ID> tuples. In the end, there were almost 747 million such tuples. To improve the selection queries on this table, we renormalized it by concatenating all concepts associated with each citation into a comma-separated list that is <citation ID; ðconcept1; concept2; . . . Þ>: In this work, we assume the data set D to be fixed. However, in practice, D changes frequently as new citations are added and existing citations are updated to include new terms from the

MeSH hierarchy.

In this case, we assume that D is refreshed periodically by an off line process that issues queries to PubMed using the concept key word and updates the concept counts and rows of retrieved citations. A newly added citation may not appear immediately in the query result.

***Online 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. This is done using the Research utility of the Entrees Programming Utilities (eUtils) [21]. EUtils are a collection of web interfaces to PubMedfor issuing a query and downloading the results with various levelsof details and in a variety of formats. Next, the Navigation treeis constructed by retrieving the MeSH concepts associated with each citation in the query result from the BioNav database. This is possible since MeSH concepts have tree identifiers encodingtheir location in the MeSH hierarchy, which are also retrieved from the BioNav database. This process is done once for each user query. The navigation tree is trivially converted to an active tree (see Section II) and passed on the Navigation Subsystem that supports the user's actions on the BioNav web interface. Initially, the navigation subsystem just visualizes the active tree on the web interface, that is, it simply shows its root node. Subsequently, the user requests an EXPAND action on the root. Then, the navigation subsystem executes the Heuristic-Reduced Opt algorithm on the tree I or Þ of the root r, and the resulting active tree is visualized on the web interface. When the user makes a SHOW RESULTS request, BioNav uses the Entrees Summary utility to download high-level information of the citations to be shown, such title and authors.

## EXPERIMENTAL EVALUATION

We evaluated the difference between the BioIntelR and BioNavsystems in terms of both average Navigation cost and expansion time performance Other traditional measures of quality, such as precision and recall, are not applicable to our scenario as the objective is to minimize the tree navigation cost and not to classify we show that the BioIntelR method, which is evaluated using middle layer and adopted BioNav system and the BioNav system Heuristic-Reduced Opt algorithm, leads to considerably smaller navigation cost for a set of real queries on the MEDLINE data base and navigations on the MeSH hierarchy. We compare the optimal algorithm (Opt-Edge Cut) with Heuristic-Reduced Opt and show that the heuristic is a good approximation of the optimal. These experiments were executed on a reduced navigation tree (_20nodes), constructed from the original query navigation tree for each query, since Opt-Edge Cut is prohibitively expensive for most navigation trees. Finally, shows that the execution time of Heuristic-Reduced Opt is small enough to facilitate interactive time use navigation. The experiments were executed on a Windows XP Professional machine with 3 GHz CPU and 2 GB of main memory, running Windows XP Professional. All algorithms were implemented in Java and Oracle 10g was used as the database.

***Navigation Cost Evaluation:***

Fig. 6, compares the Overall navigation cost of BioIntelR is Over BioNav for the biochemistry query set only. BioIntelR performs better than BioNav for all queries.

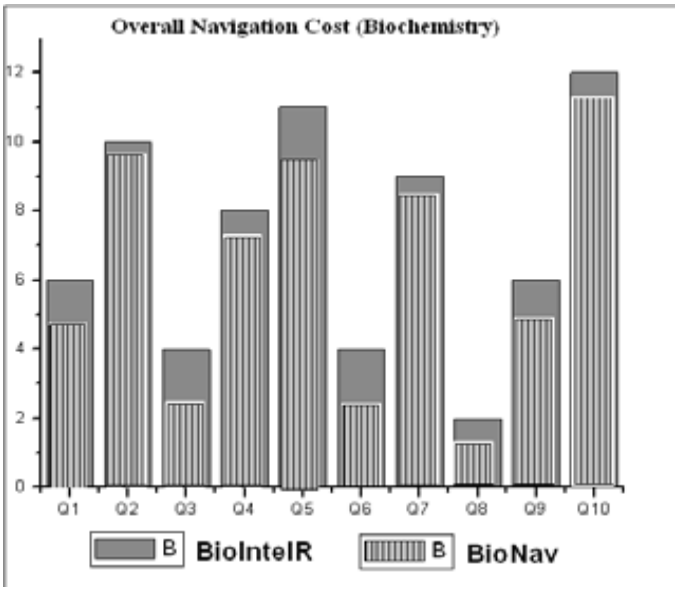



Figure 6: Overall Navigation Cost Comparison

## CONCLUSION AND FUTURE WORK

Information overload is actually a common phenomenon encountered by users searching biomedical databases such as PubMed. We address this problem by organizing the query results according to their associations to concepts of the MeSH concept hierarchy and propose a dynamic navigation method on the resulting navigation tree. Each node expansion on the navigation tree reveals a smallest of nodes, selected from among its descendants, and the nodes are selected such that the information overload observed by the user is minimized. We formally stated the underlying framework and the navigation and cost models used for evaluation of our approach. We prove that the problem of selecting the set of nodes that minimize the navigation cost is NP-complete.

## REFERENCES

[1]. K.S.N.V. Jyotsna Devi, K. RajiniKumari, "Optimizing Query Results using Middle Layers Based onConcept of Hierarchies", Data Mining, IJCST, Vol. 3, Issue 1, pp.969-974, 2012.

[2]. Abijith Kashyap, Vagelis Hristidis, M. Petropoulous, S. Tavoulari, "BioNav: Effective Navigation on Query Results of Biomedical Databases", KDD, CSE 705, Advanced Topics in database systems, 2008.

[3]. Abijith Kashyap, Vagelis Hristidis, M. Petropoulous, S. Tavoulari, "Effective Navigation of Query Results Basedon Concept Hierarchies", IEEE Transactions on Knowledge and Data Engineering, Vol. 23, 2011.

[4]. D. Lindberg, B. Humphreys, A. McCray,"The Unified Medical Language System", Methods of Information in

Medicine, Vol. 32, No. 4, pp. 281-291, 1993.

[5]. Medical Subject Headings (MeSH)(2010), [Online]Available: http: //www.nlm.nih.gov/mesh/

[6]. J.A. Mitchell, A.R. Aronson, J.G. Murk, "Gene Indexing: Characterization and Analysis of NLM's GeneRIFs", Proc. AMIA Ann. Symp., pp. 460-464.

[7]. IEEE Transactions on Knowledge and Data Engineering, Vol. 23, No. 4, April 2011.

[8]. K. Chakrabarti, S. Chaudhuri and S.W. Hwang: Automatic Categorization of Query Results. SIGMOD Conference2004: 755-766.

[9]. Z. Chen and T. Li: Addressing Diverse User Preferences inSQL-Query-Result Navigation. SIGMOD Conference 2007:641-652.

[10]. Medical Subject Headings(MeSH).http://nlm.nih.gov/mesh/

[11]. Trans insight GmbH – Go PubMed. http://gopubmed.org.

[12]. A. Kashyap, V. Hristidis, M. Petropoulos and S. Tavoulari: BioNav: Effective Navigation on Query Results of Biomedical Databases. ICDE 2009 (to appear).

[13]. L. Comtet, "Advanced Combinatory: The Art of Finite and Infinite Expansions", pp. 176-177, Redial, 1974.

[14]. Stanford Univ. (2008),"High Wire Press", [Online] Available:http://highwire.stanford.edu/

[15]. D. Maglott, J. Ostell, K.D. Pruitt, T. Tatusova, "Entrees Gene: Gene-Centered Information at NCBI", Nucleic Acids Research, Vol. 33, pp. D54-D58, Jan. 2005.

[16]. Transinsight GmbH—GoPubMed, [Online] Available: http://www.gopubmed.org/, 2008.

[17]. J.S. Agrawal, S. Chaudhuri, G. Das, A. Gionis, "Automated Ranking of Database Query Results", Proc. First Biennial Conf. Innovative Data Systems Research, 2003.

[18]. K. Chakrabarti, S. Chaudhuri, S.W. Hwang, "Automatic Categorization of Query Results", Proc. ACM SIGMOD, pp. 755-766, 2004.

[19]. M. Kaki, "Findex: Search Results Categories Help When Document Ranking Fails", Proc. ACM SIGCHI Conf. Human Factors in Computing Systems, pp. 131-140, 2005.

[20]. Entrees Programming Utilities and the applications of it http://www.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.

[21]. D. Demner-Fushman, J. Lin,"Answer Extraction, Semantic Clustering, and Extractive Summarization for Clinical Question Answering", Proc. Int'l Conf. Computational Linguistics and Ann. Meeting of the Assoc. for Computational Linguistics, pp. 841-848, 2006.