# To Extract Articles From PubMed Database Using Perl Software Tool Perl_Su

**Sugam Sharma[1], Tzusheng Pei[1], Hari Cohly[2], Raphael Isokpehi[2], and N Meghanathan[1]**

Department of Computer Science[1], Department of Biology[2]
Jackson State University, USA

**Summary**
PubMed is the suitable database for all Computational Biologists /Bioinformatics researchers to search the required data efficiently. Extraction of the data from the PubMed needs suitable input to search and careful selection the back end. A naïve user needs a good exercise before handling the PubMed database actually such as where and what input string is supplied and what database is selected in PubMed corpus. We have designed a tool using Perl script, named as Perl_Su. This tool makes the searching process easy in PubMed corpus. When we run this tool, it asks the user to supply the related string of search as input. And thus it extracts all the possible sets consisting of that supplied string. This tool is unique in its own kind and extremely useful for naïve user and will work as cantilever for those experts to develop further innovative ideas who understand the chemistry between Perl and Bioinformatics. We believe that this tool will contribute significantly in the field of Text Mining in PubMed Corpus.

***Key words:***
*Perl, Mechanize, User Agent*

## 1. Introduction

PubMed is a one of the largest database used for computational biology /bioinformatics research. Perl (Practical Extraction and Reporting Language) is also playing very import role in bioinformatics research. The increasing popularity of Perl with bioinformatics has motivated us to do this research work. We have developed a software robot system using Perl language. Software robots are the software programs which are useful to connect to the database of any website. We have focused our research work to PubMed database. The requirement of this work is Perl DOS console and Perl environment. We have downloaded the Perl from CPAN. Perl package must be consisted of PPM (Perl Package Manager). We have used certain class to a make the system automatic and the most important method is the Mechanize () which makes the systems automatic. When our tool runs it ask for the input string which the user is looking for to extract form the PubMed database. This tool extracts all the related articles of that input string from PubMed database. The results are stored is the output file placed in the bin directory of Perl Package. This file has html format and we have named it as "sugam.htm". We have collected numerous results with variety of input strings. We have divided the whole paper in sections and the rest of the paper is organized as follows.

Section 1.1 consists of the description about the Perl. As Perl description is huge to describe completely so we have emphasized on the basics of Perl. Section 1.2 is the next section under parent section introduction which consists of the detailed description about PubMed. We have touched very basic things with the help of pictorial representation with arrows dictating the meanings of the fields in the PubMed web pages. This section consists of four subsections also named as (1.2.a) Summery Format - depicts the summary of any articles, (1.2.b) Brief Format – depicts the briefings about the article, (1.2.c) Abstract Format – this format shows the abstract part of any article, and (1.2.d) Citation Format- this format provides the personal name as subject, chemical names of substances used in the research and the grant which supported this project. Section 2 is the Perl Robots Design, in which we have written all the software design code along with the proper description of every line just after that line. Next section is section 3, the Results. Section 4 is the future work which we intend to do extend after this. Section 5 is the conclusion section. And the last section is section 6 which is the reference section.

### 1.1 Perl

Perl is powerful and dynamic language knows for today's time. It is the extraction language as its name suggested "Practical Extraction and Reporting Language". These days Perl is much popular because of its coordination use with bioinformatics research. As today huge amount of biological database is available on different servers. To access this database for further analysis is the crucial task to be done. In more technical term we need to do text mining in different databases. A number of software solutions have been provided to promote the bioinformatics research and biologist are using that regularly. A good number of website and study materials are available to be able to work in Perl environment. Perl is the first great language for people which require no

Manuscript received October 5, 2007
Manuscript revised October 20, 2007

prior programming experience. Perl is the language of Unix /Linux environment and actually comes pre-installed on most non-Windows operating systems. CPAN, the Comprehensive Perl Archive Network, the main and huge library for Perl that is freely available and covers almost every fractional part of Perl programming and makes it exceedingly simple to create useful programs. In common language the Perl program are known as Perl scripts.

## 1.2 PubMed

The exact definition of PubMed lies in the form of interface. It is a public interface freely available to MEDLINE. It provides access to information in MEDLINE, the integrated molecular biology databases included in the National Center for Biotechnology Information (NCBI) Entrez retrieval system, out-of-scope citations (e.g., articles on plate tectonics or astrophysics) from certain MEDLINE journals, primarily general science and chemistry journals, for which the life sciences articles are indexed for MEDLINE. (i.e. Cell and Science) ,Citations that precede the date that a journal was selected for MEDLINE indexing, some additional life science journals that submit full text to PubMed

Central and receive a qualitative review by NLM, Citations that has not yet indexed with MeSH terms i.e. PREMEDLINE. Once you enter your search terms and click GO or press the Enter key, PubMed will automatically: Run the search, Retrieve and display citations, Retain the search terms in the query box.

Example: autism in early childhood

Number of citations found

Active query box displays current search

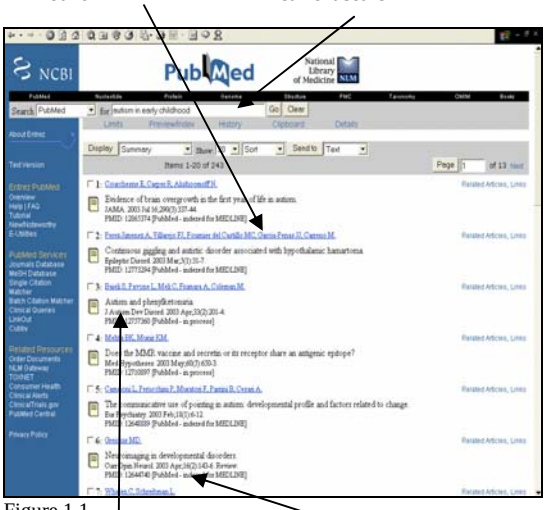

Figure 1.1

Icon indicating presence of abstract and availability in PubMed Central

Citations are displayed in a summary format. This includes author, title and source

a. Brief Format

The Brief format will display: The first author's name, the first thirty characters of the title, the PubMed unique identifier, links to related articles, and Link Out and molecular biology databases.

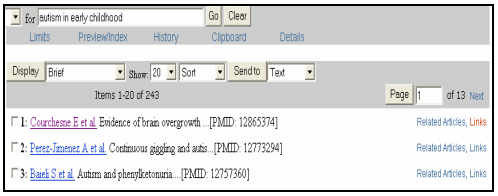

Figure 1.2

b. Summary Format

PubMed citations are displayed by default in the Summary format which consists of the following:

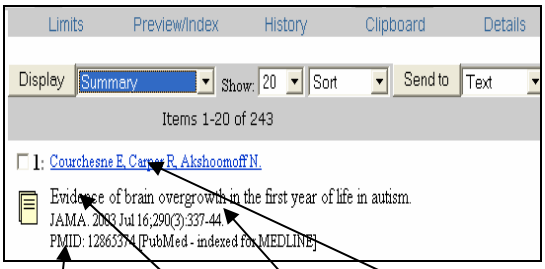


Figure 1.3

Title of article

Author's name(s)

Identification number: PubMed's unique identifier

Article source: provides the journal title abbreviations, date of publication, volume, issue and page numbers of the article

c. Abstract Format

Abstract Format provides the summary information in addition to:

First author's affiliation at the time of publication

Links to full-text article at provider's Web site, if available

Links to related articles, books, LinkOut and molecular biology databases

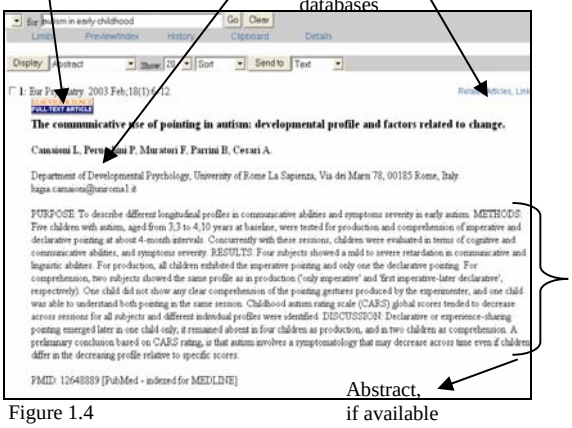


Figure 1.4

Abstract, if available

Abstract Format may also provide: Publication Types, when the type is something other than a Journal Article, Erratum, if any and Comments, if any.

d. Citation Format
The Citation Format displays the same information as the Abstract Format in addition to:

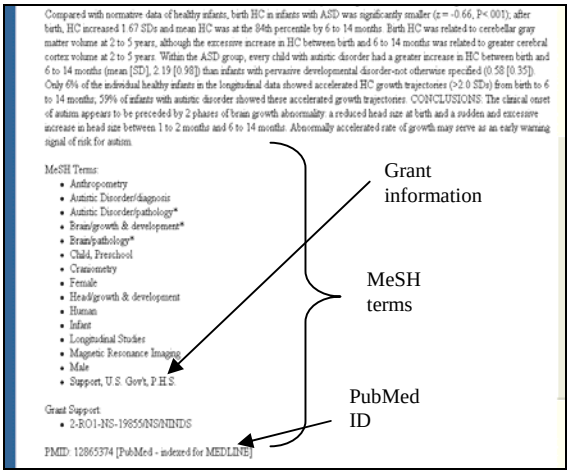

Figure 1.5

The Citation Format may also provide: Personal name as subject, if present, Chemical names of substances used in research, if present.

## 2. Perl Robot Design

In the designing of Perl Robot we have taken the following steps. Each step involved has it description just below that. Every code is ended by a semi-column.

#!c:\\perl\\bin

These are the comment points to the installation path of the PERL. This can be customized depending where the Perl directory exists.

# use strict;

Comments:

# use LWP::UserAgent;

This line is a comment as it is followed by #. LWP stands Library for WWW access in Perl. The LWP::UserAgent is a class implementing a web user agent. LWP:: UserAgent objects can be used to dispatch web requests. In normal use the application creates an LWP::UserAgent object, and then configures it with values for timeouts, proxies, name, etc. It then creates an instance of HTTP::Request for the request that needs to be performed. This request is then passed to one of the request method the UserAgent, which dispatches it using the relevant protocol, and returns a HTTP:: Response object. There are convenience methods for sending the most common request types: get(), head() and post(). When using these methods then the creation of the request object is hidden as shown in the synopsis above. The basic approach of the library is to use HTTP style communication for all protocol schemes. This means that you will construct HTTP::Request objects and receive HTTP::Response objects even for non-HTTP resources like gopher and ftp. In order to achieve even more similarity to HTTP style communications, gopher menus and file directories are converted to HTML documents.

use WWW::Mechanize;

WWW::Mechanize, is a proper subclass of LWP::UserAgent. It helps to automate interaction with a website. In short it is also known as Mech. It supports performing a sequence of page fetches including following links and submitting forms. Each fetched page is parsed and its links and forms are extracted. A link or a form can be selected, form fields can be filled and the next page can be fetched. Mech also stores a history of the URLs you've visited, which can be queried and revisited.

my $url = "http://www.pubmed.org";

Any variable in Perl is define using $ before any variable name. If it is to made local variable we use ‘my’ word before the variable declaration.

Print "Enter the String"."\n";

This line prints the string.

chomp($var = <>);

This line take is input and store that in a variable and to avoid to jump into next line chomp is used.

my $searchstring = "$var";

Value from one variable is assigned to another local variable.

my $outfile = "sugam.htm";

Output html file named as “sugam.htm” is assigned to a local variable.

my $mech = WWW::Mechanize->new();

This creates a new WWW::Mechanize object and stores a handle to the object in the variable that we called $mech.

```
$mech->get($url);
```

This causes the WWW::Mechanize object to go out and fetch the page that we are requesting.

```
$mech->follow_link(text => "PubMed",   n => 1);
```

Follows a specified link on the page. You specify the match to be found using the same parms that follow_link() uses. Here the first link called is PubMed.

```
$mech->form_name('EntrezForm');
```

Name of the form where the values are directed.

```
$mech->field ('SearchBar.Term' => "$searchstring");
```

Name of the field on that form in which the values are actually submitted.

```
$mech->click ();
```

This provides the environment for button click, as if you were interacting with the page yourself.

```
my $output_page= $mech->content();

open (OUTFILE, ">$outfile");
print OUTFILE "$output_page";
close (OUTFILE);
```

The above four line are doing the collaborative task like assigning the content of the returned page to $out_put page, open a simple output file, write the contents to the file, and then close the file.

## 3. Result

We have collected different sets of results. We have taken the snap shots of the output based on the supplied input. The screenshot contains the outcome and the right top corner of each figure consists of the dos console where we run the software. It is utmost important to pay attention on the directory of output file. In the URL section in the screenshot we have save the out file in C drive and the path is C://Perl/bin/sugam.htm.

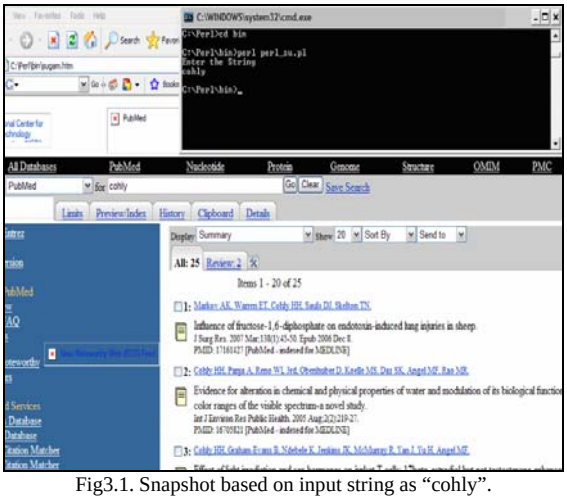

Fig3.1. Snapshot based on input string as "cohly".

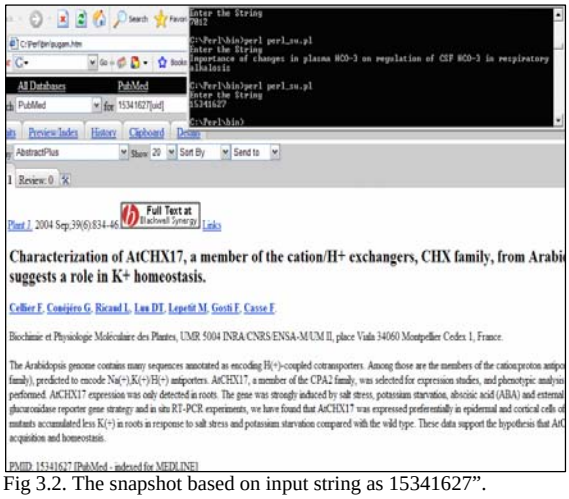

Fig 3.2. The snapshot based on input string as 15341627".

This figure 3.2 shows the output as "Characterization of AtCHX17 ….." This article has the PMID number as 15341627 which we supplied in the input and got that output shown in the figure.

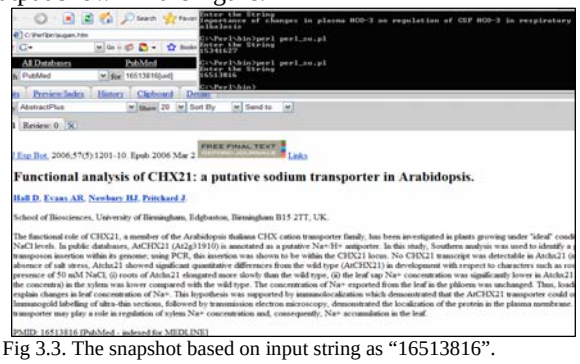

Fig 3.3. The snapshot based on input string as "16513816".

This figure 3.3 shows the output as "Functional Analysis of AtCHX21: a putative sodium transporter in Arabidopsis." This article has the PMID number as 16513816which we supplied in the input and got that output shown in the figure.

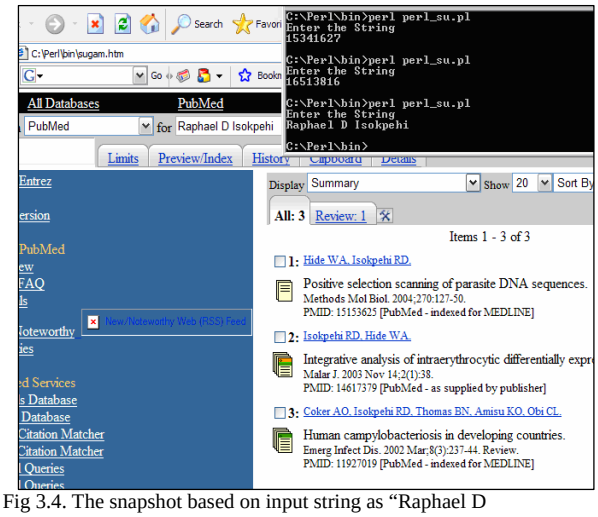


Fig 3.4. The snapshot based on input string as "Raphael D Isokpehi".

This figure 3.4 shows the possible output from PubMed database with respect to the supplied input as Raphael D Isokpehi.

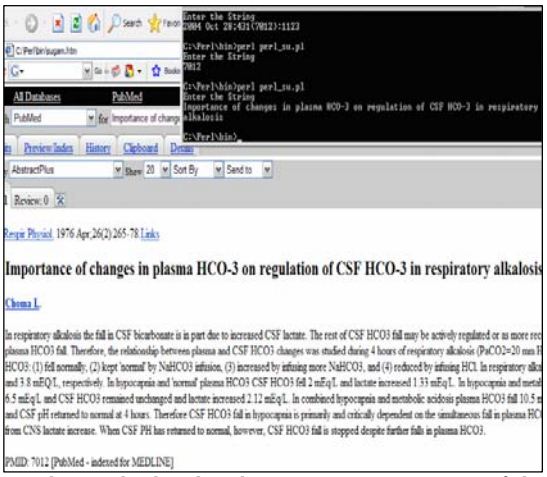


Fig 3.5. The snapshot based on the input string as "Important of changes in plasma HCO-3 on regulation of CSF HCO-3 in respiratory alkalosis."

This figure shows the abstract view of the article which consists of the title as "Important of changes in plasma HCO-3 on regulation of CSF HCO-3 in respiratory alkalosis." And this title we have used as the input string in our software.

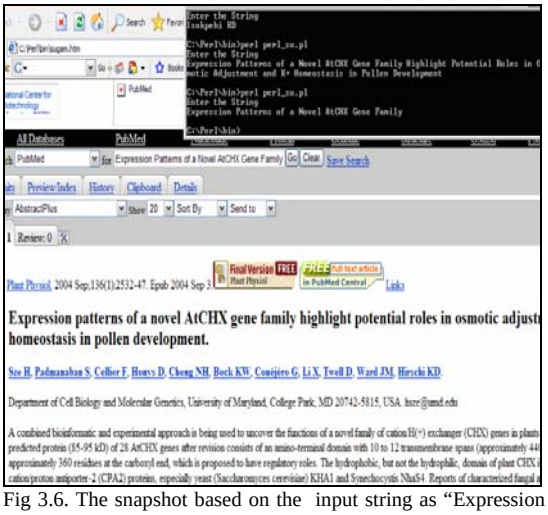


Fig 3.6. The snapshot based on the input string as "Expression patterns of a novel AtCHX gene family highlight potential roles in osmotic adjustment and K+ homeostasis in pollen development

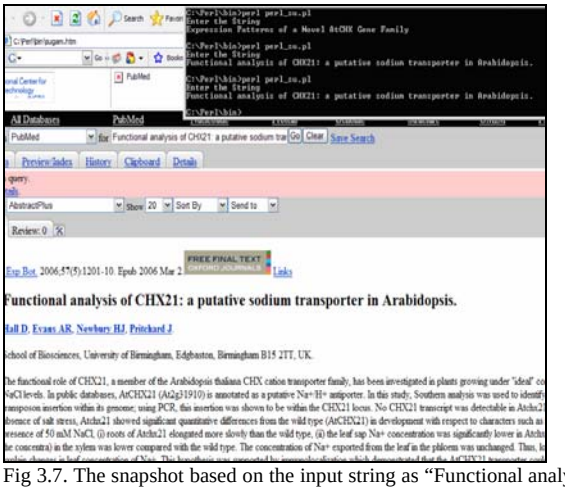


Fig 3.7. The snapshot based on the input string as "Functional analysis of CHX21….. ."

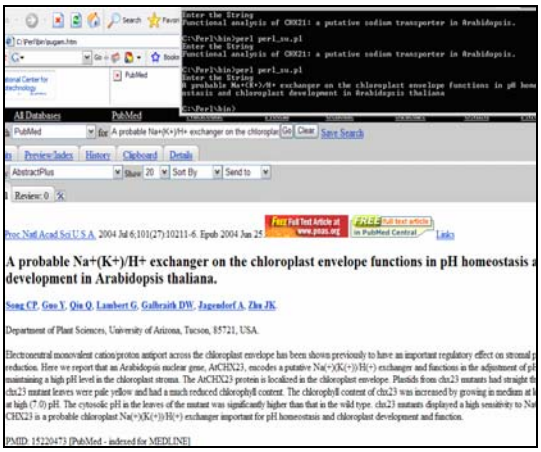


Fig 3.8. The snap shot based on the input string as " A probable Na+(K+)/H+ exchanger on the chloroplast envelope functions in pH homeostasis ……"

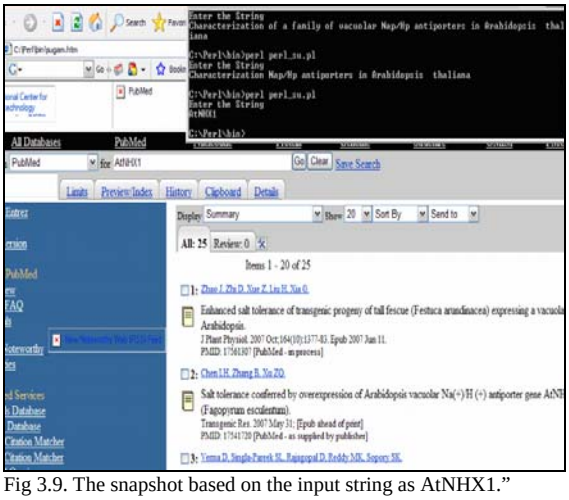

Fig 3.9. The snapshot based on the input string as AtNHX1."

## 4. Future Work

So far we have restricted our software for the PubMed database. Though this is the strongest database but there are other databases al well which have utmost important for bioinformatician and researchers. We plan to extend our work in the same direction but extend that to different databases such as Blackwell synergy etc. which consist of a huge collection of useful data for researchers and scientist. As Perl script is heavily used for bioinformatics so we want to strengthen this belief too. Though many more solutions and researches are possible with the tuning combination of Perl and Bioinformatics we would like to make significant contribution in that arena in future.

## 5. Conclusion

We all know that Perl script is an extremely suitable language for the bioinformatics. Our work is one more strong evident of the same. We have shown how Perl robots are able to fetch the data from the PubMed database. We have tested our robot using different input strings and we have happy to say it is working efficiently for all of them. Our robot so far is working with PubMed /Medline database but we are interested to extend our task to different databases to make the system more flexible for computational biologists and researchers. We believe that this research work will be able to contribute significantly and open avenues for new research.

## References

[1] www.pubmed.org
[2] http://cdd.unm.edu
[3] www.cpan.org