

# Apellicon: a web-based tool for constructing and curating Textpresso databases

William M. Urbanski<sup>1, 2</sup> and Brian G. Condie<sup>1</sup>

<sup>1</sup>Department of Genetics

<sup>2</sup>Department of Computer Science

University of Georgia

Athens, GA 30602

## Abstract

As more research literature in the biological sciences is made available in electronic format, text mining systems are increasingly being used to improve the ability of investigators to retrieve relevant information. Through the use of advanced indexing techniques that utilize biological ontologies, semantic databases, and other formal representations of biological concepts text mining systems have been able to effectively parse biological literature. While text mining systems are increasingly effective at creating the linkages required to provide context-specific search results, the systems themselves are difficult to set up and use by novice computer users due to the highly technical nature of the applications. Because most researchers in the biological sciences do not have a strong computer science background we have focused on improving the quality of existing, proven text mining systems by implementing a web-based GUI that greatly improves the workflow of these systems. Textpresso in particular has an excellent web-based interface for searching literature but does not have an easy to use administrative interface. We developed the Apellicon interface to enable a wide range of users to build and manage a Textpresso database. An important feature of Apellicon is that it can enable groups to collaborate in building a Textpresso database.

## Introduction to Textpresso

Textpresso is a text-mining system for the bioscience literature (Muller et al., 2004). There are a number of Textpresso web servers that have been developed by groups in a variety of fields (International Arabidopsis Informatics Consortium, 2010; Gama-Castro et al., 2008; Garten and Altman, 2009; Harris et al., 2010; Muller et al., 2008; Shen-Orr et al., 2009; Skrzypek et al., 2010; Urbanski and Condie, 2009; Yamazaki et al., 2010). Textpresso uses ontologies to index sentences contained within papers to enable the discovery of linkages between concepts that may not be explicitly expressed in the body or abstract. Textpresso excels at providing a search interface that allows the user to query the database with search terms that reveal ideas expressed in the literature that were identified through the ontologies. Textpresso parses scientific literature in PDF or XML format and generates a web interface for searching the

## **Apellicon: a tool for constructing Textpresso databases**

parsed literature. Because of this dichotomy it is best to think of Textpresso as combination of two applications. One application is a collection of scripts that parses scientific literature in text format and builds indices based on a set of user specified ontologies. The other component is a web application that provides a web interface to search through the parsed text. While the Textpresso search interface is intuitive and provides access to the search system the interfaces provided for building a Textpresso database are challenging for novice users to implement.

The Textpresso system is written entirely in Perl, uses a file system based database, and runs in a Linux environment. The web application component of Textpresso will run on most Perl-enabled Linux web servers (our installations use Apache 2.0). While the web interface can easily be accessed and utilized by any novice computer user with access to a web browser, the scripts that parse the literature and build the database require a knowledge of and access to the Linux command line environment.

Installing Textpresso is straightforward. After downloading Textpresso from [textpresso.org](http://textpresso.org), extract the Textpresso scripts from the archive and run the installation script. The installation script will help prepare your system and will configure most aspects of the Textpresso application. After Textpresso has been installed you can begin to build your database. To begin you must first convert your source literature to text format. A Linux program called pdf2text comes with the Textpresso package and can convert literature in PDF to plaintext. Next the bibliographic information for the articles must be made available to Textpresso. This is done by populating a special set of directories on the Textpresso server with the appropriate bibliographic information. Each directory corresponds to a particular type of bibliographic data; the author directory contains information about the paper's authors, while the body directory contains the full text of the publication. While Textpresso provides some scripts that can automatically build a database they are mainly geared towards querying PubMed for certain search terms and automatically building the database with the list of returned results. There is no convenient or accessible way to manually curate the results received from PubMed, as they are automatically loaded into the database. This is convenient for users who wish to quickly build a database without manually selecting the publications. However, in some cases it is desirable to build a Textpresso databases from collection of selected publications. For example in building the publication database for the Textpresso Site Specific Recombinase web server (<http://ssrc.genetics.uga.edu/>) (Urbanski and Condie, 2009) we found that conventional PubMed searches designed to find publications describing uses of Cre recombinase returned a very large number of irrelevant papers. This was due to the retrieval of many publications focused on the transcription factor CREB and/or its binding site Cre by entering the keyword "Cre" into PubMed. Apellicon allowed us to build a Textpresso

## Apellicon: a tool for constructing Textpresso databases

database from a group of publications that had been manually screened for relevance thereby eliminating a large number of publications that would not be of interest to the user.

Once the bibliographic directories have been populated the Textpresso database is created by running the process source files script. This script creates indices of the literature based on the ontologies configured in the system and generates the searchable webpages available through the web browser. Most of the process behind building a Textpresso database requires a knowledge of the Linux shell as well as a knowledge of how the Linux system and Apache webserver are configured. While Textpresso does make the installation process simple with its installation script the process of building a complete and curated database is still difficult for novice users due to the complexity of the system. We have simplified and automated the process of importing literature and building the database by developing a web application called Apellicon. Apellicon builds on Textpresso by providing a novel, intuitive web interface for building Textpresso publication corpora.

### A description of the Apellicon interface

Apellicon uses a web-based GUI to import PubMed bibliographic data into a Textpresso database (Figure 1). For each PubMed ID entered Apellicon will retrieve the information that populates the abstract text, accession number (PubMed ID number), author name(s), citation information, journal name, title, publication type and year. In the text area labeled *PubMed ID*: the user can input a list of PubMed ID numbers to be downloaded and imported to the



The screenshot shows the Apellicon web interface. At the top, there is a logo consisting of a stack of books and the word "Apellicon". Below the logo are navigation links: "Home", "Job Viewer", and "About". A teal-colored link "site-specific-recombinase-tools" is also visible. The main heading is "Import Bibliographic Data" with a "(Help)" link. Below this is a large text input area labeled "PubMed ID:". Underneath the input area are radio buttons for "Format" with "PMID" selected and "PMCID" as an option. Below the format options is a "Delimiter:" dropdown menu currently set to "(space)". At the bottom of the form are three buttons: "Import", "Create Stub Only", and "Cancel".

Textpresso database as a batch.

Multiple delimiter types are supported so ID numbers stored in spreadsheets, flat files, or other existing data sources can easily be imported. The 'Create Stub Only' option can be used to generate blank entries in the database if the user intends to revisit the entry later and manually insert bibliographic data.

After importing the PubMed IDs of the article(s) into a Textpresso database there are two options for importing the corresponding PDF files of the papers. If the PDF is available online Apellicon will automatically download and parse

**Figure 1. The Apellicon interface for importing PubMed bibliographic data into a Textpresso database.**

## **Apellicon: a tool for constructing Textpresso databases**












the article and insert it into database. If the PDF is not available, the user can upload the document from their local system via their web browser. In this case the user is presented with a simple 'Browse' dialog box to select the document from their local system and upload it through their web browser. Once the document has been uploaded it is automatically converted to text format and put in the appropriate directory on the Textpresso server. The user receives a confirmation that the process has completed and their document has been added. Alternatively the PDF files can be uploaded in bulk through FTP/SFTP. This option can be used if the user needs to import a large number of papers from their local computer. Files uploaded to Apellicon in this way can then be uploaded into Textpresso via the Import PDFs page.

The Textpresso scripts process a large amount of information during the generation of a database; as a result they require a significant amount of time to execute. When executed on the command line these programs slow the build process by forcing the user to wait for the scripts to finish before performing other actions on the database. To speed up the process of database construction Apellicon uses an internal task scheduler to execute the Textpresso scripts. The status of the tasks that have been scheduled can be viewed via the Job Viewer interface in Apellicon (Figure 2). When an action is performed in the web interface like downloading bibliographic data from PubMed or importing a PDF file, Apellicon generates the appropriate job to be executed by the task scheduler. Scheduling jobs instead of performing them in real time allows the Apellicon web interface to be responsive even while the scripts are executing and forces the jobs to be executed in the background. Additionally if many jobs are pending execution many instances of the task scheduler can run in parallel, speeding up the build process. While these enhancements are available to Textpresso users through the use of advanced Linux commands, Apellicon implements these enhancements by default and never requires the user to wait while a process is executing, minimizing the amount of time required to work with a Textpresso installation. To make troubleshooting easier the job viewer allows system administrators to see exactly which scripts Apellicon is running in the background. If the system is experiencing problems administrators have the ability to manually run and debug these scripts without having to guess what Apellicon is doing.

Many aspects of the Textpresso application require that only one user work on a given database at a time. For instance, it is not possible to have multiple users generating the database and adding new literature to the system at the same time. Apellicon eliminates conflicts by providing a multi-user interface for building Textpresso corpora. Multiple users can work on a single Textpresso installation concurrently without the fear of overwriting or destroying another user's data. Because of Apellicon's task scheduler all actions with the Textpresso system are executed in first in, first out (FIFO) order allowing multiple users to add literature, generate bibliographic data, and rebuild the database concurrently.

## Job Viewer

 Remove old jobs

Submitted	Status	Command
2009-10-07 12:07:30		/usr/local/textpresso/tcell/Procedures/scripts/ProcessSourceFiles.pl
2009-10-07 12:03:57		./bin/pdf_to_tokenized_text.pl /var/www/tools/pdf_import /var/www/tools/temp/pdf_proc /usr/local/textpresso/tcell/Data/includes/body
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/ab5a1c1212_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/25c09b8bfc_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/f800e09c5a_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/44fb6fb766_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/600ce54061_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/96a17e0cc7_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/c608ce8982_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/d08c419e20_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes
2009-10-07 12:01:41		./bin/get_bib_info.pl ./temp/job/7b295adc30_textpresso_import.pmidfile /usr/local/textpresso/tcell/Data/includes

**Figure 2. The Apellicon job viewer.**

The Apellicon job viewer provides a detailed glimpse into the inner workings of the Textpresso build process. Because Apellicon relies heavily on an internal task scheduler to automate the build process the Job Viewer allows system administrators to more easily troubleshoot build issues by seeing the exact commands that Apellicon is executing on the system.

In addition to these features Apellicon includes a Data Summary interface allowing users to monitor the contents of a Textpresso database (Figure 3). The Data Summary interface lists each publication in the database by PubMed ID number. Green check marks indicate that the indicated fields within the Textpresso database entry for that publication are complete, while red Xs indicate that the field is not complete. Clicking on the magnifier icon in a cell gives the user access to a text box containing the contents for that field. In cases where information is missing (abstracts from older publications are often not listed in the PubMed record) the missing content can be manually inserted into that field from a different source. This feature allows incomplete database entries to be completed by manual information entry.

## Data Summary [\(Help\)](#)

Internal ID	abstract	accession	author	body	citation	journal	title	type	year	Complete?
12704203										
16917506										
17433324										
17242199										
17998203										
18250451										

**Figure 3: The Apellicon Data Summary interface.** Each reference in the Textpresso database is listed by its PubMed ID number (Internal ID). The status of each field in the database is indicated by a green check for complete fields or red Xs for incomplete fields. Clicking on the magnifier icon leads to a form for manually entering or correcting the information in that field.

## **Apellicon: a tool for constructing Textpresso databases**

In summary, Apellicon's web-interface is intuitive and allows computer users who are unfamiliar with the Linux command line to build Textpresso corpora using any web browser without any knowledge of the underlying structure or functionality of Textpresso. Researchers can get more value out of their Textpresso installations via collaborative database construction and curation facilitated by Apellicon. Apellicon also allows easy collaborative maintenance and updating of a Textpresso database.

Apellicon has been written in PHP 5 and is released under the GNU General Public License version 3.0. It has been tested and is compatible with most Textpresso-compatible web servers. Apellicon has been designed to work with Textpresso 2.0 and implements interfaces for the Alere 1.1 batch processing scripts that are available on the Textpresso website. Apellicon requires the MySQL database engine.

### **Availability and requirements**

<i>Project Homepage</i>	<a href="http://apellicon.sourceforge.net">http://apellicon.sourceforge.net</a>
<i>Supported OS</i>	Textpresso-compatible Linux & Unix
<i>Other Requirements</i>	PHP 5 MySQL 5
<i>License</i>	GNU General Public License version 3

### **Acknowledgements**

The authors thank Hans-Michael Muller and Ruihua Fang of the Textpresso group at The California Institute of Technology for their advice. This work was supported by The University of Georgia.

## **Literature Cited**

International Arabidopsis Informatics Consortium (2010). An International Bioinformatics Infrastructure to Underpin the Arabidopsis Community. *Plant Cell In Press*.

Gama-Castro, S., Jimenez-Jacinto, V., Peralta-Gil, M., Santos-Zavaleta, A., Penaloza-Spinola, M.I., Contreras-Moreira, B., Segura-Salazar, J., Muniz-Rascado, L., Martinez-Flores, I., Salgado, H., *et al.* (2008). RegulonDB (version 6.0): gene regulation model of *Escherichia coli* K-12 beyond transcription, active (experimental) annotated promoters and Textpresso navigation. *Nucleic Acids Res* 36, D120-124.

Garten, Y., and Altman, R.B. (2009). Pharmspresso: a text mining tool for extraction of pharmacogenomic concepts and relationships from full text. *BMC Bioinformatics* 10 Suppl 2, S6.  
Harris, T.W., Antoshechkin, I., Bieri, T., Blasiar, D., Chan, J., Chen, W.J., De La Cruz, N., Davis, P., Duesbury, M., Fang, R., *et al.* (2010). WormBase: a comprehensive resource for nematode research. *Nucleic Acids Res* 38, D463-467.

Muller, H.M., Kenny, E.E., and Sternberg, P.W. (2004). Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biol* 2, e309.

Muller, H.M., Rangarajan, A., Teal, T.K., and Sternberg, P.W. (2008). Textpresso for neuroscience: searching the full text of thousands of neuroscience research papers. *Neuroinformatics* 6, 195-204.  
Shen-Orr, S.S., Goldberger, O., Garten, Y., Rosenberg-Hasson, Y., Lovelace, P.A., Hirschberg, D.L., Altman, R.B., Davis, M.M., and Butte, A.J. (2009). Towards a cytokine-cell interaction knowledgebase of the adaptive immune system. *Pac Symp Biocomput*, 439-450.

Skrzypek, M.S., Arnaud, M.B., Costanzo, M.C., Inglis, D.O., Shah, P., Binkley, G., Miyasato, S.R., and Sherlock, G. (2010). New tools at the *Candida* Genome Database: biochemical pathways and full-text literature search. *Nucleic Acids Res* 38, D428-432.

Urbanski, W.M., and Condie, B.G. (2009). Textpresso site-specific recombinases: A text-mining server for the recombinase literature including Cre mice and conditional alleles. *Genesis* 47, 842-846.

Yamazaki, Y., Akashi, R., Banno, Y., Endo, T., Ezura, H., Fukami-Kobayashi, K., Inaba, K., Isa, T., Kamei, K., Kasai, F., *et al.* (2010). NBRP databases: databases of biological resources in Japan. *Nucleic Acids Res* 38, D26-32.