

# An Interactive User Interface for Drug Labeling to Improve Readability and Decision-Making

Hamed Abedtash<sup>1</sup>, PharmD, Jon D. Duke, MD, MSc<sup>2</sup>

<sup>1</sup>School of Informatics and Computing, Indiana University, Indianapolis, IN;

<sup>2</sup>Regenstrief Institute, Indianapolis, IN

## Abstract

*FDA-approved prescribing information (also known as product labeling or labels) contain critical safety information for health care professionals. Drug labels have often been criticized, however, for being overly complex, difficult to read, and rife with overwarning, leading to high cognitive load. In this project, we aimed to improve the usability of drug labels by increasing the 'signal-to-noise ratio' and providing meaningful information to care providers based on patient-specific comorbidities and concomitant medications. In the current paper, we describe the design process and resulting web application, known as myDrugLabel. Using the Structured Product Label documents as a base, we describe the process of label personalization, readability improvements, and integration of diverse evidence sources, including the medical literature from PubMed, pharmacovigilance reports from FDA adverse event reporting system (FAERS), and social media signals directly into the label.*

## Introduction

Food and Drug Administration (FDA) prescribing information (also known as product labeling or labels) are primarily meant to be used by physicians and other healthcare professionals to find accurate, reliable drug information needed to prescribe medicines safely and effectively. The labeling provides evidence-based information on adverse reactions, contraindications, dosing modifications in children and the elderly, and guidance regarding use in pregnancy and nursing<sup>1</sup>. However, labels have also been extensively criticized for having too much information, making them complex, lengthy, and challenging to readers seeking relevant and important safety warnings<sup>2-4</sup>. In our previous quantitative analysis<sup>3</sup>, we found the “overwarning” problem with the adverse drug events (ADE) in the labeling content which hinders clinicians from grasping the significant safety alerts in the labels.

When reviewing a drug label, health-care providers must scan the entire label content to find relevant safety warnings, a cumbersome and time consuming task. While the introduction of label Highlights has brought some of this information to the fore, these facts are limited to the most severe or frequent events in the overall population, without consideration of patient-specific factors. Providers utilizing labels are most commonly seeking a particular piece of information related to a particular patient, whether based on co-morbidities, co-medications, or symptoms that have been raised in a visit. Finding such information with a label is inefficient and often impossible within the time constraints of a typical office visit. There are, of course, myriad websites and applications that provide drug safety information. However, the FDA-approved label is unique—it is thoroughly vetted and approved by a standardized process, and is recognized by manufacturers as the defining statement regarding their drugs. Thus, working with the actual label to improve its usability is a necessary and novel advancement in providing a better experience for providers while maintaining the ‘blessing’ of manufacturers and the FDA.

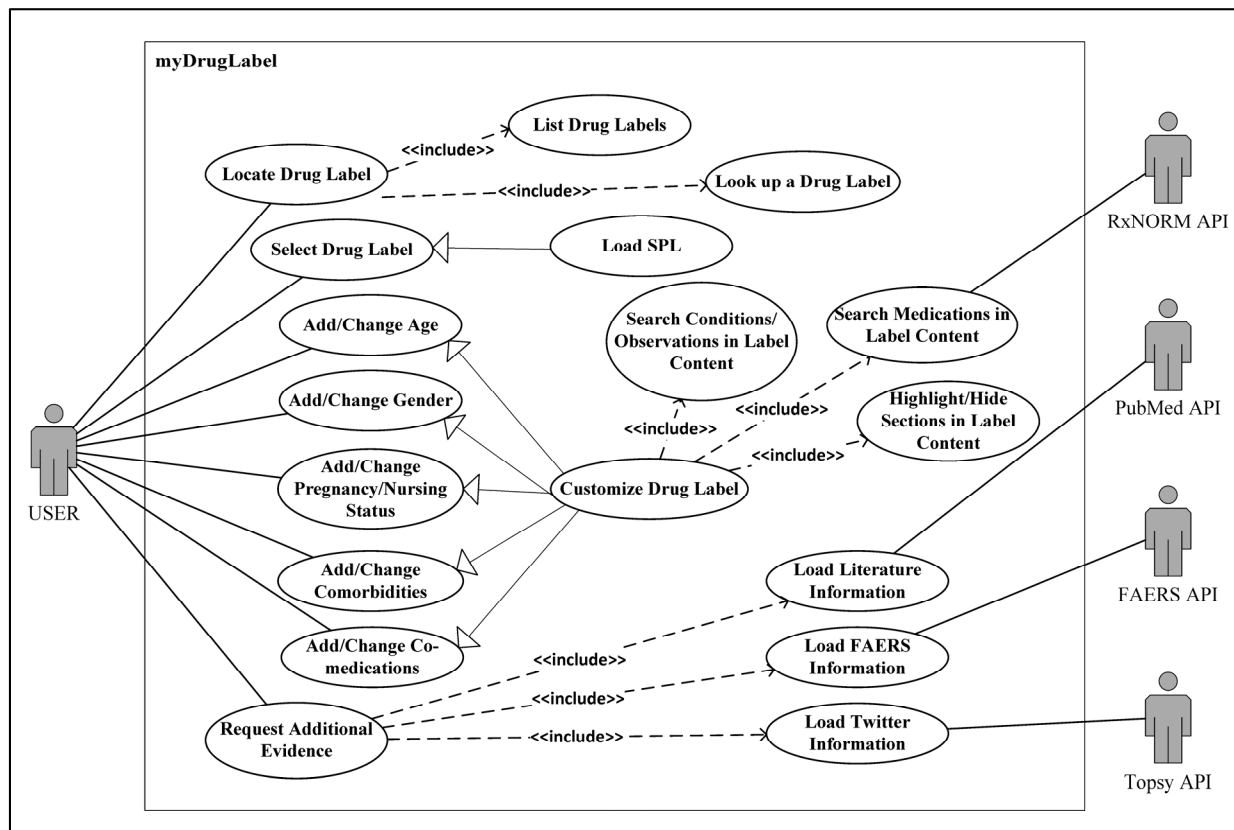
In this paper, we describe the development of an interactive, web-based application known as myDrugLabel. The goals of this project were to 1) improve the readability of FDA-approved drug labels by increasing the ‘signal-to-noise ratio’, 2) provide meaningful information to the physicians based on patient-specific comorbidities and concomitant medications, and 3) support the decision-making process by integrating additional evidence sources (e.g., medical literature, pharmacovigilance reporting, social media signals) directly into the label. In the following sections, we will provide an overview of the application design process and walk through the resulting application’s architecture and features through a sample clinical scenario.

## Methods

### *Requirements and Design*

We applied an iterative design process to prototype, develop, and refine the application components including data processing algorithms, user interface, and database. In each iteration, the developed functionalities were tested and necessary modifications were made. Healthcare professionals, e.g., physicians, pharmacists, and nurses were identified as target users of the application with a possible extension to patients as we add enhanced usability.

In the first step, the functional requirements of the application were elicited through a group composed of one physician and two pharmacists (including authors HA and JD) to define use cases and task scenarios. Figure 1 represents the use cases defined by this group, including label searching, modification of patient-specific factors, and integration of data from multiple sources. The requirements called for five data ‘providers’, the user and four external Application Program Interfaces (APIs), i.e. RxNORM, PubMed, FDA Adverse Event Reporting System (FAERS), and Topsy. The diagram represents the functional requirements for the system and the interactions between actors (e.g., user) and use cases (e.g., search a drug label). The “include relation” indicates that the included use cases are part of a larger use case and executed upon parent use case is called. The “use relation” or large hollow triangle arrow shows that the process of doing a use case (e.g., Customize Drug Label) always involves executing another use case (e.g., Add/Change Age).



**Figure 1.** Use case diagram of myDrugLabel application. The “include relation” specifies that the use cases are executed as part of a parent use case, and the “use relation” or large hollow triangle arrow indicates that execution of the base use case involves performing another use case. API=application program interface, FAERS=FDA adverse event reporting system, SPL=structured product labeling.

In the next phase, prototypes of the user interface were developed based on these requirements. Both designs had data entry sections on the left side; however, the design with a simpler interface in which supplementary information was displayed in tooltips above the highlighted terms was felt to be a better experience by the test users. The prototypes were evaluated by external users, and changes were made to the prototype interface based on the comments.

#### Technology Platforms

The proposed architecture was based on standard medical terminologies, open-source software and database platforms, and publicly available application programming interfaces (APIs). RxNORM<sup>5</sup> and MedDRA<sup>6</sup> (Medical Dictionary for Regulatory Activities) were used to map drug names and adverse events within the drug labels,

respectively. HTML5, CSS3, JavaScript, jQuery library, and AJAX were used to develop interactive webpages with asynchronous data transfer capability. Data were requested and transferred via PHP (server-side language for web application development) APIs from a MySQL database in extensible markup language (XML) or JavaScript Object Notation (JSON) formats.

### *Structured Product Labeling (SPL)*

Structured Product Labeling (SPL) is the standard format accepted by FDA for transferring drug labeling contents between information systems. FDA drug labels were retrieved in SPL format from DailyMed and transformed to HTML files using the latest SPL schema available on FDA website, retrieved on October 14, 2013<sup>7</sup>.

### *Processed SPL (P-SPL) Documents*

In order to improve the performance of filtering label contents by diseases and comorbidities, a Python application was developed that identified adverse event terms in the SPL based on MedDRA hierarchy using simple natural language processing (NLP) and regular expression techniques to encapsulate in a span HTML tag. This approach saved time and decreased CPU load for the client facing interface. For example, the term ‘myalgia’ in the SPL was converted to `<span class="condition" mdrptid=" 36516837">myalgia</span>`. This modification allowed for rapid manipulation of the visual design of the label based on user input, without needing to search for terms in real-time (i.e., user waiting time). The response time highly depends on the client system configuration, length of the drug label, and how many times the term is mentioned in the label content. In average, the modification reduced the user waiting time from 60 to 2 seconds on a Desktop PC with CPU i5-3470 3.3 GHZ and RAM of 4MB.

### *Supplemental Evidence and Information*

One of the recognized challenges to interpreting adverse events in drug labels is the lack of external evidence. For example, is an event listed simply because it was reported to the FDA, or have there been substantive studies exploring the associated risk between drug and event? We thus sought to supplement the labels with additional external evidence sources, displayed in a tooltip after selecting a drug adverse reaction or drug name. This was accomplished through the use of APIs pulling in data from the literature (PubMed API<sup>8</sup>), the FDA adverse event reporting system (FAERS) (openFDA API<sup>9</sup>), and Twitter posts (Topsy API<sup>10</sup>). The information was streamed via respective APIs in a real-time manner.

FAERS data were retrieved from openFDA platform launched in June 2014. The queries to the API comprised two sections. In the case when an adverse reaction is selected on the label, the label’s drug name along with the MedDRA preferred term of the reaction are concatenated. Throughout our preliminary testing the API, we learned that both generic name and brand name of the drug should be in the query to fetch all records. For example:

```
https://api.fda.gov/drug/event.json?search=patient.drug.medicinalproduct:atorva
statin+lipitor+AND+patient.reaction.reactionmeddrapt.exact:myalgia
```

Likewise, when a drug name on the label is selected, the chosen drug name and the drug label’s name are sent to API. The returned counts do not necessarily indicate the number of drug-drug interaction events. For example:

```
https://api.fda.gov/drug/event.json?search=patient.drug.medicinalproduct:atorva
statine+lipitor+AND+patient.drug.medicinalproduct:digoxin
```

## **Results**

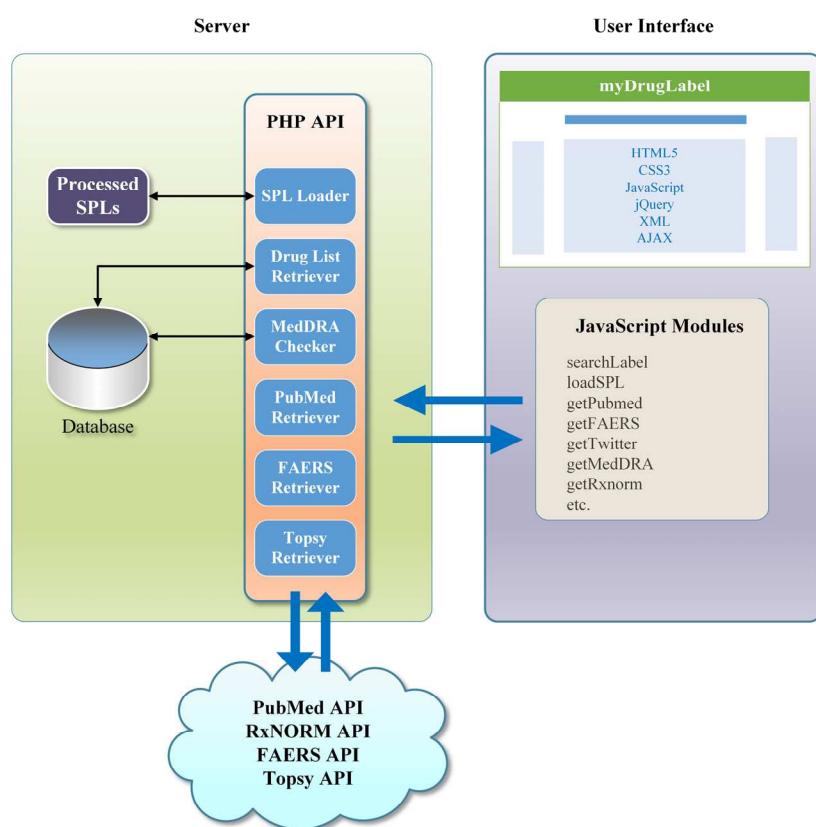
### *Application Architecture*

The architecture of myDrugLabel is a Service Oriented Architecture (SOA) design in which the client- and server-side components are connected via RESTful web services with the server database and external APIs (Figure 2). The client-side includes webpages, JavaScript codes, and P-SPL document. The server-side component is composed of a database, the bank of P-SPL documents, and PHP APIs.

The application has two pages, Home and Label View. The user starts looking up a drug name on Home page to select a drug of interest. Then, the page navigates to Label View page to display the labeling content and filtering functionality. In order to maximize speed of the client-side user interface, all data transfer processes between the server and browser are done in an asynchronous fashion. The only exception is when the drug labeling content is retrieved from the server. On the Home page, a request is sent asynchronously to the API once the user searches a

drug name, then the top 10 drug names similar to the input string are returned to fill the drop down list. Selecting the drug name directs user to the Label View page, which is the main page for processing the labeling content. The selected drug's P-SPL document is loaded from server via the SPL Loader API and gets transformed to an HTML file by using the FDA-provided XSLT schema.

The application does not have a stand-alone business layer. Part of the calculations and text processing are performed on the client browser by JavaScript functions, including highlighting or hiding content sections, terms, displaying notifications on the right side, and tooltip functionality. Some other processes that needed security constraints, such as database connections and external API data transfer were designed to be performed by the server-side PHP APIs. The server APIs include SPL Loader (fetches SPLs from server by the requested SPL ID), Drug List Retriever (gets drugs lists from the database), MedDRA Checker (returns comorbidities' synonyms from MedDRA dictionary located in the database), PubMed Retriever (requests and processes the literature statistics from PubMed), FAERS Retriever (requests and returns ADE statistics from FAERS), and Topsy Retriever (requests, calculates, and returned the number of signals on Twitter).

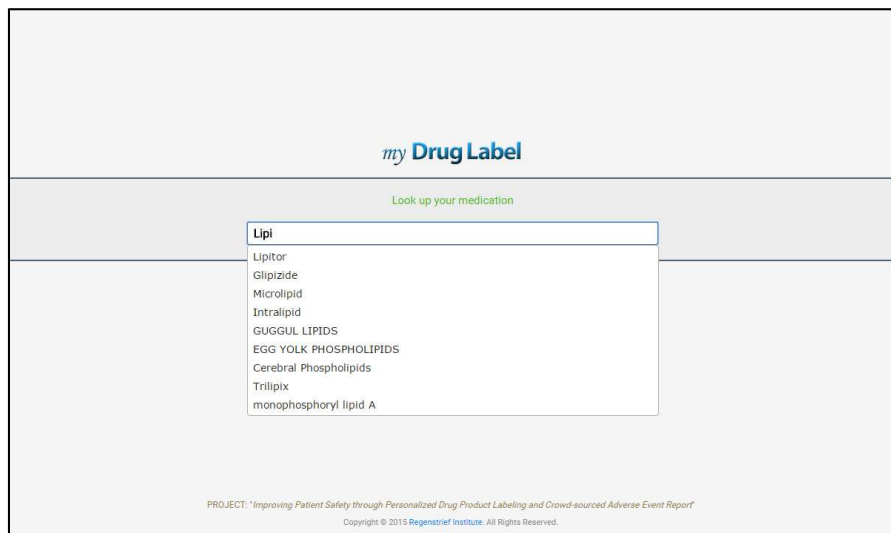


**Figure 2.** Simplified view of myDrugLabel architecture. AJAX=asynchronous JavaScript and XML, API=application program interface, FAERS=FDA adverse event reporting system, MedDRA=medical dictionary for regulatory activities, PHP=server-side language for web application development, SPL=structured product labeling.

### Application User Interface

Once a drug is selected from the Home page (Figure 3), the page is directed to Label View page (Figure 4), which is the primary location for interactivity in the application. On this page, the user can enter patient-specific parameters, which dynamically alter the labeling content (Figure 4, section A). For example, if the patient's age is greater than 65 years old, the geriatrics section is automatically highlighted and the pediatrics sections is become hidden. Similarly, upon selection of male gender, the pregnancy and nursing sections are hidden. Making these adjustments enhances the readability of the label by reducing unnecessary content. Similarly, when a patient condition is added,

all synonyms of the disease are retrieved via the MedDRA API and highlighted in the label text (Figure 4, section B). Highlighting of conditions is limited to the following safety-related sections: Adverse reactions, boxed warnings, clinical studies, contraindications, warnings and precautions, general precautions, other safety information, pregnancy, risks, use in specific populations, and user safety warnings.



**Figure 3.** Home page of myDrugLabel application. User searches for a drug name in the provided textbox; once a drug is selected, it directs to Label View page.

To the immediate right of the label content is a customized navigation bar (Figure 4, section C) that provides quick access to highlighted sections of the label (based on patient parameters, such as pregnancy information or geriatric dosing instructions). The navigation bar also provides rapid access to highlighted terms within the label, such as relevant conditions, medications, and adverse effects. To help the user keep track of this content, related concepts are color coded throughout both the content and the navigation areas. Thus, for instance, if the user enters a condition of ‘Myalgia’, all information related to the condition will be highlighted in the same color throughout the document. Myalgia-specific highlights can then be toggled on and off as a group. For certain sections, color-coding is applied in standard fashion for all labels. For example, the pregnancy and nursing sections are highlighted in pink.

Integrated evidence from external data sources was shown via on-demand tooltips, for better accessibility and to avoid information overload. The tooltips are accessible by clicking on any highlighted term, either comorbidities or co-medications. The tooltip is composed of four tabs (Figure 5), including an evidence overview, literature data, pharmacovigilance reporting data, and social media signals. The overview tab gives the user a general sense of the strength of association between the selected term (comorbidity or co-medication) and the drug of interest. On the literature tab, the statistics of published papers mentioning both the drug (generic and brand) and the condition or its synonyms are presented stratified by type of publication, including journal article, case reports, clinical trials, controlled clinical trials, meta-analyses, observational studies, randomized controlled trials, and review articles. The link to PubMed search page is also provided, which will take the user to a new window containing the PubMed search results specific for that drug, condition, and article type.

The ADR reports tab displays how often the comorbidity or co-medication has been reported by other consumers of the drug of interest. These reports are divided into categories based on seriousness: 1) congenital anomaly, 2) disabling, 3) hospitalization-associated, 4) life-threatening, 5) death, 6) all serious events, and 7) all non-serious events. The social media tab provides real-time data on mentions of the drug and condition of interest (or its synonyms) on Twitter. The number of posts are grouped by 1-day, 7-day, 30-day, 6-month, and 1-year time frames.

#### *Usage Scenario*

An example scenario for myDrugLabel is demonstrated in Figure 4. A 71-year old woman is taking Lipitor for her hypercholesterolemia. She has also other comorbidities including diabetes, myalgia, hypertension, and heartburn for

which she is taking acetaminophen, metformin, hydrochlorothiazide, and cimetidine. After the patient data is entered, the sections related to geriatrics turn to green, while pediatrics, pregnancy care, and nursing sections are hidden. The numbers next to the comorbidities and co-medications on the filter and notification bars show how many times they are mentioned in the label's content. As shown in the figure, the sections that contain the terms are highlighted such as warnings, precautions, adverse events, and clinical trials, making it easy to find the relevant information specific to a particular co-morbidity or symptom.

The screenshot displays the 'my Drug Label' interface for 'LIPITOR- atorvastatin calcium trihydrate tablet, film coated'. On the left, a filter box (A) allows users to input age (71 years), gender (Woman), and various conditions (Diabetes, Hypertension, Myalgia, Heartburn) and medications (Metformin, Hydrochlorothiazide, Acetaminophen, Cimetidine). The main content area (B) shows the drug label text, with sections like '8.5 Geriatric Use' highlighted in green. On the right, a notification bar (C) lists sections that are highlighted (e.g., Geriatric Use) and hidden (e.g., Pregnancy, Nursing Mothers, Pediatric Use). A navigation bar (D) is located at the bottom right of the content area.

**Figure 4.** The Label View page. Using the filter box (A), user is able to customize the label content (B) by entering age, gender, pregnancy and nursing status, comorbidities and co-medications on the left side. On the right side (C), the user is notified of found sections and terms related to the input data. The navigation (D) bar helps user jump quickly to the point of interest within the content.

### Discussion and Future Directions

In this project, we designed, prototyped, and developed an interactive user interface for FDA-approved drug labeling called myDrugLabel, reflecting the “personalized” aspect of the application. This web-based application promises improvement in readability of drug labeling and support for evidence-based decision-making. A formal evaluation study of the tool is underway. However, it has several novel features in its architecture and visual design. First, it utilizes the original FDA-approved product label as a base for all enhancements, injecting personalized highlighting

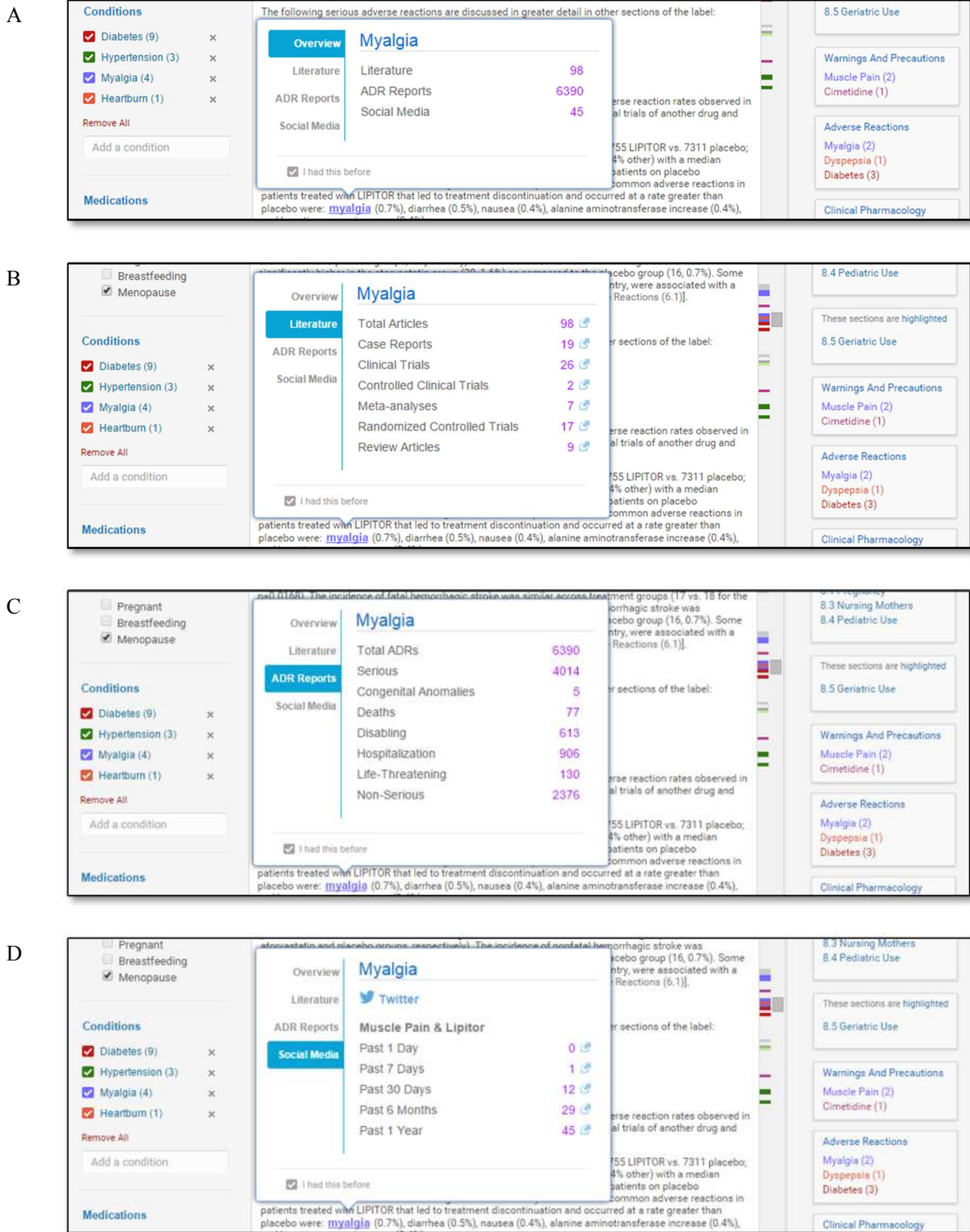
and multiple external evidence sources into the original labeled content. As a result, the user can review the original label, exactly as FDA approved, or by entering a few patient parameters, review a shorter, focused document highlighting the specific areas of interest to the practitioner.

MyDrugLabel incorporates a number of existing APIs to pull real-time data from evidence sources, then further stratifies this information to maximize usability. Most critically, myDrugLabel assists the provider in determining whether a labeled side effect, potentially one of hundreds listed, is in fact supported by other sources. This knowledge is critical to informing both the patient and prescribing decisions.

At the time of developing the website, the SPL information were accessible on the National Library of Medicine's Download Labels site (DailyMed). In August 2014, openFDA released drug product label API (<https://open.fda.gov/drug/label/>) that provides access to over 25,000 prescription and 36,000 over-the-counter (OTC) structured product labeling records. The new API has three main advantages. First, SPL information can be retrieved with high level of granularity, meaning that the user has the option to download pieces of the label, e.g., Boxed Warning, Adverse Reactions, Drug Interactions, Patient Counseling Information, etc. Second, it is possible to perform simple SPL mining using the API, for instance, to find the medications that have a specific ingredient or adverse reaction. Third, the API has the power to perform a descriptive analysis, e.g., to identify the number of products having a specific food interaction with alcohol or grapefruit juice. Considering these capabilities, the new iterations of myDrugLabel will be a proper showcase of applying openFDA drug product label API.

A limitation of this paper is the lack of usability metrics to quantify the benefits experienced by users of this tool. However, many of myDrugLabel's features, such as simply removing pregnancy information from labeling for male patients, shows the intuitive benefits of its design. Another limitation is that the presence of abstracts in the literature or mentions on Twitter do not necessarily imply a causal or direct association between drug and effect. The search strategies for these sources need to be refined to better interpret the evidence for users of the application.

Several next steps are planned for the development and evaluation of the application. First, as mentioned, we will evaluate the application with healthcare professionals from different specialties as well as patient users. Secondly, we will incorporate evidence from observational data networks to provide more clinically sound evidence of associations between drug and adverse events. Finally, myDrugLabel will be released for public use, both to gather feedback on larger scale and to incorporate patient-reported adverse effects into the evidence base.



**Figure 5.** The tooltip element provides supplementary information about conditions and co-medications. A) The Overview tab shows total number of items in each evidence section group. B) The Literature tab presents a stratified count of papers in the literature citing both the drug of interest and the selected term. C) The ADR Reports tab shows the number of reported cases associated with the drug and selected condition. D) The Social Media tab provides the count of Twitter posts mentioning the specific term (or its synonyms) and the drug of interest.



## References

1. Food and Drug Administration, HHS. Requirements on content and format of labeling for human prescription drug and biological products. Final rule. Fed Regist. 2006;71(15):3921-97.
2. Avorn J, Shrank W. Highlights and a hidden hazard--the FDA's new labeling regulations. N Engl J Med. 2006;354(23):2409-11.
3. Duke J, Friedlin J, Ryan P. A quantitative analysis of adverse events and "overwarning" in drug labeling. Arch Intern Med. 2011;171(10):944-6.
4. Woosley RL. Drug labeling revisions—guaranteed to fail? JAMA. 2000;284(23):3047-9.
5. U.S. National Library of Medicine. RxNorm 2014 [cited 3/11/2015]. Available from: <http://www.nlm.nih.gov/research/umls/rxnorm/>.
6. International Conference on Harmonisation of Technical Requirements for Registration of Pharmaceuticals for Human Use (ICH). Medical Dictionary for Regulatory Activities 2014 [cited 3/11/2015]. Available from: <http://www.meddra.org/>.
7. U.S. Food and Drug Administration. Structured Product Labeling Resources 2013 [cited 3/11/2015]. Available from: <http://www.fda.gov/ForIndustry/DataStandards/StructuredProductLabeling/default.htm>.
8. Sayers E. E-utilities Quick Start. 2008 Dec 12 [Updated 2013 Aug 9]. In: Entrez Programming Utilities Help [Internet]. Bethesda, MD: National Center for Biotechnology Information (US). Available from: <http://www.ncbi.nlm.nih.gov/books/NBK25500/>.
9. U.S. Food and Drug Administration. openFDA 2014 [cited 3/11/2015]. Available from: <https://open.fda.gov/>.
10. Topsy Labs, Inc. Topsy API 2015 [cited 3/11/2015]. Available from: <http://api.topsy.com/doc/>.