# Mining Officially Unrecognized Side effects of Drugs by Combining Web Search and Machine Learning


Carlo Carino
Politecnico di Milano
Piazzale Leonardo 32
Milano (MI), 20133, Italy
curino@elet.polimi.it

Yuanyuan Jia
Dept. of Computer Science, UIC
851 South Morgan
Chicago, IL 60607-7053
yjia@cs.uic.edu

Bruce Lambert
Dept. of Pharmacy Administration, UIC
833 South Wood St.,
Chicago, IL 60612-7230
lambertb@uic.edu

Patricia M. West
Dept. of Pharmacy Practice, UIC
833 South Wood St.,
Chicago, IL 60612-7230
pwest@uic.edu

Clement Yu
Dept. of Computer Science, UIC
851 South Morgan
Chicago, IL 60607-7053
yu@cs.uic.edu



## ABSTRACT

We consider the problem of finding officially unrecognized side effects of drugs. By submitting queries to the Web involving a given drug name, it is possible to retrieve pages concerning the drug. However, many retrieved pages are irrelevant and some relevant pages are not retrieved. More relevant pages can be obtained by adding the active ingredient of the drug to the query. In order to eliminate irrelevant pages, we propose a machine learning process to filter out the undesirable pages. The process is shown experimentally to be very effective. Since obtaining training data for the machine learning process can be time consuming and expensive, we provide an automatic method to generate the training data. The method is also shown to be very accurate. The side effects of three drugs which are not recognized by FDA are validated by an expert. We believe that the same approach can be applied to many real life problems and will yield high precision. Thus, this could lead a new way to perform retrieval with high accuracy.



## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *selection process;* I.2.6 [**Artificial Intelligence**]: Learning –*Connectionism and neural nets.*

## General Terms

Algorithms, Measurement, Experimentation.

## Keywords

Mining side effects of drugs, machine learning, accurate retrieval, precision.


## 1. INTRODUCTION

All medications have both benefits and risks. In the United States, drug companies conduct time-consuming and expensive clinical trials of new drugs before they are marketed to the public. The results of these studies are reviewed by the U.S. Food and Drug Administration (FDA). A drug is allowed to enter the market only if the FDA determines that its benefits outweigh its risks. Unfortunately, clinical trials, no matter how carefully they are conducted, cannot identify all potential problems [AH03]. Some events are simply too rare to be detected in trials that include, at most, a few thousand patients. Other risks become apparent only when certain kinds of patients take the drugs (e.g., children, pregnant women, people with multiple chronic problems, people taking other medications). These special categories of patients are often excluded from clinical trials, so their first exposure to the drug comes after it has been approved and marketed [LAWH02]. The FDA and the drug industry are well aware that pre-approval clinical trials routinely fail to detect significant safety problems and adverse drug effects [AH03]. A recent well-known example is Vioxx, which causes heart problems for certain patients, and the drug has to be withdrawn from the market [Couz04]. In fact, 13 drugs were withdrawn from the market by the US FDA between 1997 and 2001 [http://www.fda.gov/fdac/features/2002/chrtWithdrawals.html].

The main purpose of this work is to develop a Web-mining system that can find online evidence of side effects in approved drugs that are not yet officially acknowledged by the FDA or the drug manufacturers. In this paper, we introduce techniques which can find the officially unrecognized side effects of drugs. We argue in the conclusion that the techniques introduced here can be applied to various other problems requiring high precision retrieval. The main contributions of this paper are

(a) We provide techniques for mining side effects of drugs, and our experimental results demonstrate that the techniques are extremely effective. This is a real life problem which has an impact to millions of people.

(b) Since the mining process involves training data which can be time consuming and expensive to obtain, we provide an automatic process to obtain the desired training data and show that this process is highly effective.



*CIKM'05*, October 31-November 4, 2005, Bremen, Germany.

(c) We believe the same techniques, possibly with minor modifications, can be applied to various real life problems. This could lead to a new paradigm for high effective retrieval.

The paper is organized as follows. In Section 2, the problem of mining side effects from the Web is defined. In Section 3, we describe the components of our system to solve the problem. In Section 4, experimental results are shown to demonstrate that our approach is very promising. The side effects of three drugs which are not recognized by FDA are validated by an expert. The conclusion is given in Section 5.

### 1.1 Related Research

Some research works on mining unrecognized side-effects of drugs have been reported[ HBL98, PBLLOE02]. There are some differences between our work and the earlier works. (1) The data used in [HBL98, PBLLOE02] are hospital data and are structured data, while we mine from unstructured text data. Clearly, the techniques employed are very different. (2) While patient data may be more reliable than unstructured Web data (note that PubMed which contains huge amount of bio-medical literature is accessible from the Web), the amount of Web data on side-effects of drugs is likely to be significantly more. Both types of data should be complementary for this type of research. It should be noted that government regulations (HIPAA) may prevent researchers from using hospital patient data for research (This applies to many hospitals including UIC.)

There are some well-known web sites, such as www.yellowcard.gov.uk and www.drugs.com, which provide possible side-effects of drugs. The former site collects information of suspect side effects for quite a few drugs, which is based on case reports submitted by registered health professionals and patients. The latter site is one of the mining resources in our system, while we will try to incorporate the former site into our system.

The solution we propose for the drug side effect mining problem involves learning from training data. This bears similarities to quite a few traditional information retrieval problems such as classification [LSCP96, YL99, NMTM99], routing [Harm95, Harm96] and relevance feedback [SALT89, BYRN99, van79]. However, there are significant differences. Classification is a process where there are a number of classes, each containing a set of example documents and each new document is to be classified into one of these given classes (or a new class). Routing may be considered as a type of classification where each class is defined by a query (or a set of queries) and each new document is routed to a class. This involves sending new documents to appropriate queries, which are given while our task is to retrieve documents for new queries. Relevance feedback can be roughly classified into two types. The first type consists of manually identifying some relevant and irrelevant documents and then modifying the query (possibly using machine learning algorithms) to retrieve more relevant documents for the same query. Our approach differs from it in two aspects. First, it is possible for our method to produce positive and negative examples automatically so as to avoid the manual identification process. Second, our method applies captured relationships among terms to new queries, not to the same query. The second type of feedback, known as pseudo-feedback, assumes the first few retrieved documents as relevant and utilizes them as training data. While the assumed relevant documents in a pseudo-feedback process are chosen in a rather naïve manner (i.e. always pick the top n documents, for some n), the process of automatic generation of training examples in this paper is more elaborate. Although pseudo-feedback usually yields an improvement in average retrieval effectiveness, a significant deterioration for some queries is generally observed, as usually a substantial number of the first few retrieved documents are irrelevant. Similar to the traditional feedback process, the pseudo-feedback process is applied to the same initially submitted queries, not to different queries. Although training for some queries and then applying their results to other queries has been attempted before [YuMi88], the improvement in retrieval effectiveness for the new queries is relatively small. In contrast, the improvement due to the technique reported here is substantially higher.

## 2. PROBLEM DEFINITION

The problem is to find the side effects of drugs which have been approved by the Food and Drug Administration (FDA). For each such drug, FDA keeps a list of known side effects. Our task is to identify unknown side effects of FDA approved drugs from the Web, where **an unknown side effect is one which is not listed on the official FDA pages**.

## 3. OUR APPROACH

### 3.1 System Overview

One way of finding unknown side effects of a drug is to submit a query of the following form to a search engine such as Google.

< drug name, side effect >

And then from the retrieved pages find occurrences of symptoms/diseases. The Food and Drug Administration (FDA) keeps a list of side effects for each drug. If a symptom or a disease of the drug found in the retrieved page is outside the set of side effects of the drug as reported by FDA, then it is a potential unknown side effect of the drug. However, the above approach has the following problems:

(a) There are quite a few relevant pages which cannot be retrieved by the above query.

(b) Many pages returned by the search engine are not relevant. For example, a sentence such as "It is safe to use drug X for a patient with disease Y" in a retrieved page might give the false impression that drug X causes disease Y.

(c) As a page can sometimes be very large and too time consuming to read, it may be desirable to extract an important passage for each retrieved relevant page. (Some recent retrieval tasks already extract the answers in response to a query, see for example, the Question-Answering track of TREC [Voor01-04].)

We now sketch our system which attempts to accomplish the above tasks. It should be noted that determining the relevance of a retrieved page automatically is a difficult task. Because if the relevance of a page could be determined, the search engine would retrieve it if it were relevant, and discard it, otherwise. However, for our problem, there is additional information we can utilize. Specifically, we can semi-automatically determine the relevance of retrieved pages for a set of drugs. First we mine the characteristics of the words (features) in these pages, which make a page relevant or irrelevant. These characteristics are then applied to determine the relevance of retrieved pages of other drugs.

We now describe the components of the system as shown in Figure 1.

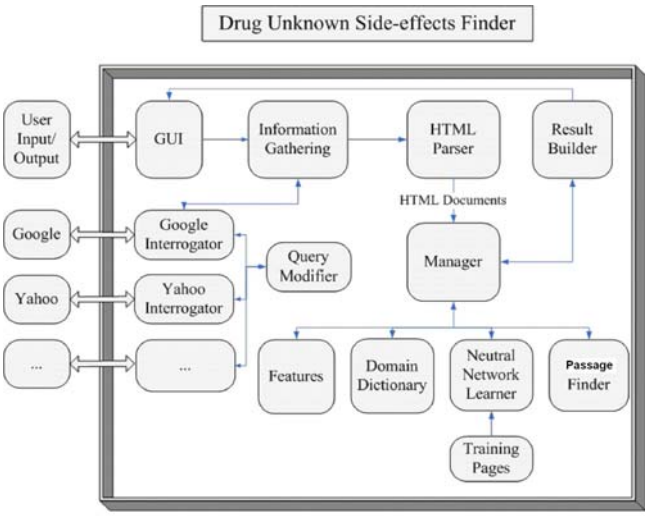


**Figure1. The architecture of the system**

## 3.2 Graphic User Interface

A GUI accepts a drug name and a user-specified value *n*, which is the number of pages to be returned by the search engine. If the search engine retrieves fewer than *n* pages, then our system will process the retrieved pages, otherwise only the top *n* pages returned by the search engine will be processed.

## 3.3 Query Modifier

A query modifier takes the drug name, goes to a FDA site to find the active ingredients of the drug (which are the chemical compounds forming the drug) and then forms the query

< drug name > OR < its active ingredients >

The query is then submitted to Google. We can also submit the query to other search engines such as Yahoo and PubMed.

More and possibly better pages are returned when the active ingredients are added, as some relevant pages do not have the drug name but have its active ingredients only. A Web page which describes the side effects of a drug may not contain the actual phrase "side effect" (for example, it may contain the phrase "adverse reaction"). A query containing the phrase "side effect" and the drug name may also retrieve pages unrelated to the drug. For these reasons, we decide to drop the phrase "side effect". To reduce response time, the file containing the active ingredients of drugs from FDA is actually loaded into our system so that an access to a local file is sufficient.

## 3.4 HTML Parser

Currently only HTML pages will be analyzed. Each page is parsed and the text data are extracted. In the future, pages in other formats will be analyzed.

## 3.5 Domain Dictionary

A dictionary containing adverse effects such as nausea, vomiting and liver damage is utilized. A small synonym dictionary is also included. This allows us to detect, for example, that side effects and adverse effects have the same meaning. The advantage is that our system can be used for different applications. Specifically, if the application is changed, then a different set of domain dictionaries will need to be supplied. For example, if we are interested in finding the advantages of cell phones, the synonym dictionary may contain phrases which have the same or similar meanings as cell phones, while another dictionary may contain various potential advantages such as convenience and portability.

## 3.6 Features

Some features which can be used to determine whether a page can be useful are as follows:

(a) Presence of diseases/symptoms;

(b) Distance between certain keywords such as "side-effect" and a disease/symptom;

(c) Absence of expressions such as “no side-effects” or “safe”.

Currently, about 250 features are used. It is feasible to use all features and then apply feature selection techniques ( say based on information gain [HaKa00]) to eliminate unnecessary features. Distance features in (b) help to strengthen the casual relationships between drugs and side-effects. Note that if the distance between “side effect” and a disease is small, then the disease is likely to be caused by the drug, while if the distance between “safe” and a disease is large, then the drug may not cause the disease.

### 3.7 Training Data

The training data are a set of retrieved pages and their relevance/irrelevance for a set of drugs. Each page in the set of training examples is manually determined to be relevant or irrelevant. In the experiments, 223 pages from 7 drugs: Prozac, Aspirin, Paxil, Celebrex, Beclomethasone, Namenda are used as training examples.

### 3.8 Learner

Quite a few machine learning methods can be used for this project and our impression is that there will not be drastic differences in accuracy for using one machine learning method versus another. In our experiment, we employ a neural network, Lamstar neural network [GK98]. The neural network can handle imprecise input data and even missing data. It is briefly described as follows.

The network consists of several (23 in our experiments) sections in an input layer of neurons (which are self organizing maps) and 1 layer of output neurons. Associated with each input section of neurons is a set of closely related features. For example, one input section of neurons can be associated with the features which represent the phrases “side effects”, “adverse effects” and “adverse reactions”. Another input section of neurons may be associated with a set of symptoms/diseases. The neurons in a section represent combinations of values of the features. For example, in the section for the phrases “side effects”, “adverse effects” and “adverse reactions”, a neuron may represent “001”, indicating the absence of the first two phrases and the presence of the third phrase, while another neuron may represent “000”, indicating the absence of all three phrases. When a page is retrieved, all values of the features will be extracted from the page and used as inputs to the neural network. Consider an input section, L, of neurons and the impact of the inputs (restricted to the features of the neurons in L) on those neurons. Either the inputs are close (as measured by some distance function, for example, input 011 may be considered to be close to the neurons representing 001) to the combinations of feature values as represented by some of the neurons in L or they are far away from those feature values as represented by all neurons in L. In the former cause, the neuron in L whose feature values are closest to the input is fired. In the latter case, a new neuron which represents the inputs (restricted to L) is added to L and is fired. In general, the entire input causes at most one neuron in each input section to fire. (If no input data is missing, then exactly one neuron in each input section is fired.)

The output layer consists of two neurons only for this application (although in general there can be more neurons). The $i$th neuron ($i$=1 or 2) in the output layer is connected to the $j$th neuron in the $k$th input section as represented by a binary variable $x(i, j, k)$. When the $j$th neuron in the $k$th input section fires, $x(i, j, k)$ =1, $i$ varying over the 2 neurons in the output layer; otherwise it is 0. In our application, the firing of the neuron causes $x(1, j, k)$ = $x(2, j, k)$ =1, as we have two output neurons. However, the firing of the neuron may have different impacts on the two output neurons as quantified by the weights $w(1, j, k)$ and $w(2, j, k)$. In the training phase of the neural network, these weights $w(1, j, k)$ and $w(2, j, k)$ are automatically determined by the network, based on the relevance/irrelevance of the pages. In the testing phase, i.e. determining if a retrieved page is relevant or not, the network computes:

$$S1 = \sum_{j,k} w(1, j, k) \text{ and } S2 = \sum_{j,k} w(2, j, k),$$

where $k$ ranges over all input sections and for each section, only one neuron (identified by $j$) per section (identified by $k$) is fired. Suppose the first neuron in the output layer denotes relevance and the second one denotes irrelevance. Then the two sums, $S1$ and $S2$, are compared. The larger one of the two sums causes the corresponding output neuron to fire. Thus, a page is determined to be either relevant (containing side effects) or irrelevant. The bigger the difference between the two sums, the more confident the neuron network has in making the determination. Since each of the two sums is computed by firing input neurons, and each input neuron is associated with the values of a set of features, it is easy to find out which feature values are significant in determining the relevance or irrelevance of a given page. A generalization would have three neurons in the output layer, representing severe side effects, mild side effects and no side effects.

The neural network has a mechanism to determine whether a set of feature values as represented by a neuron in an input section are significant or not. Let the neuron be the $j$th neuron in the $k$th section. If $w(1, j, k)$ is approximately equal to $w(2, j, k)$ after the training phase, then the contribution of the neuron towards determining the relevance of the pages is immaterial and therefore that specific neuron is not useful. Consider a feature $f$ in the $k$th input section of neurons for $f$ and a set $S$ of other features associated with the $k$th section. There are two sets of

weights $\sum_j w(1, j, k)$ and $\sum_j w(2, j, k)$ where each sum is over all neurons having identical values of the features in *S* but different values of *f*. If for each sum of weights in the former set, the corresponding sum in the latter set has approximately the same value, then the feature *f* has essentially no impact in determining the relevance of a page and therefore *f* can be safely deleted. Currently, we do not prune any feature; in the future, useless features can be pruned to yield a more efficient system. The difference between the two sums can be used to indicate the significance of the features: the larger the difference, the higher significance the feature is.

### 3.9 Passage Finder

When a page is determined by the neural network to be relevant, we are interested in finding the "most important" passage within the page. A window of a fixed number of words is initially set at the beginning of the page. The window is adjusted to make sure that complete sentences are included in the window. Within this window, the number of occurrences of the most significant features, namely the symptoms/diseases and phrases "side effects" or its synonyms are counted, while the other features such as "safe" are ignored. Then the sum $\sum_i f_i * g_i$, where $f_i$ is the number of occurrences of the *i*th significant feature and $g_i$ is the degree of importance of the feature, is computed.

The window is then moved to find the next passage and the above computation of sum is repeated. The passage which yields the largest sum is the passage which is considered most important for the page. An example passage extracted using our algorithm is shown below.

"…headache with stiff neck severe nausea or vomiting yellowing of eyes or skin side effects that usually do not require medical attention (report to your prescriber or health care professional if they continue or are bothersome): constipation or diarrhea difficulty swallowing dizziness gas or heartburn minor upset stomach nausea or vomiting what should *i* watch for while taking rofecoxib? (back to top) let your prescriber or health care professional know if your pain continues; do not take with other pain-killers without advice. If you get flu-like symptoms (fever, chills, muscle aches and pains), call your prescriber or health care professional; do not treat yourself to reduce unpleasant effects on your stomach. …". It is likely that by examining this extracted passage of the page, a human can determine whether the page contains information about the side effects of the drug.

### 3.10 Result Builder

For each page which our system judges to be relevant, we extract the URL of the page so that a human can examine whether the page (or at least the extracted passage of the page) contains side effect information. Furthermore, the names of the side effects in the page are recorded. In addition, the number of occurrences of each disease is kept. The higher the number of times a side effect is mentioned the more likely that the drug causes the side effect. The Web may contain some erroneous materials. This step attempts to eliminate these erroneous materials by not taking into consideration side effects with very low frequencies of being mentioned. The side effects are arranged in descending number of times they were mentioned.

### 3.11 Finding Unknown Side-effects

We relied on two sites for known side-effects. One site is

http://www.fda.gov/cder/index.html

where the name of the drug is submitted to find the official side effects of the drug. Here, technical medical terms are used. Another site

http://www.nlm.nih.gov/medlineplus/druginformation.html

provides side-effects of drugs for consumers. Ordinary non-medical terms are used in that site. If a side-effect of a drug determined in the steps given above is outside the two sets of side-effects of the drug in the two sites, then it can be considered as a officially unrecognized side-effect of the drug.

## 4. EXPERIMENTS

### 4.1 Experimental Setup

We first describe the setup of our experiments. Initially, queries involving 7 arbitrarily chosen drugs as described in Section 3 are submitted to Google, with each query representing a drug. The 7 drugs are Prozac, Advil, Aspirin, Paxil, Celebrex, Beclomethasone and Namenda. For each drug, the number of pages retrieved by Google is limited to 100. Each retrieved page related to a drug is examined manually by a user to identify whether it describes a side-effect of the drug. This forms the training data. The training data are fed to the neural network to obtain the various weights associated with the input neurons and the two output neurons. Then, for each drug in a set of drugs which have the empty intersection with the initial set of drugs, a query is formed and is submitted to Google. Each page retrieved by Google is fed into the neural network which classifies the page to be relevant (containing some actual side effect of the drug) or irrelevant (does not contain any side effect of the drug). Clearly such a classification needs not be accurate and will need to be verified manually.

We apply the classification technique to a set of 20 other drugs ( Vioxx, Meridia, Crestor, Accutane, Serevent, Bextra, Lipitor, atenolol, Synthroid, Premarin,

Zithromax, Caduet, Avastin, Vidaza, Ketek, Sanctura, Apokyn, EstroGel, Alimta, Campral). The first 6 drugs were mentioned in recent FDA drug safety hearings; the next 5 drugs were most frequently prescribed in 2003 and the remaining drugs are approved by FDA in 2004.

We show the side effects of these drugs as mined from the Web. We note that these results are obtained purely from a statistical point of view and actual verification of whether these side effects are reliably associated with exposure to the drugs needs to be carefully examined by medical experts. Finally, we recognize that manually obtaining the training data is time consuming. Thus, we propose the following technique to construct training examples automatically. We perform experiments to find out the accuracy of the following process to determine positive and negative examples of training data automatically.

(1) Identify manually a set of drugs from FDA sites such that each such drug has a set of known side effects, say *S1* and a set of known diseases it is used to treat, say *S2* with the constraint that the intersection of *S1* and *S2* is empty.

(2) Submit a query of the form <d, s2> to PubMed or Google, where d is one of the drugs given in step (1) and s2 is a disease in *S2* of the drug d.

(3) For each retrieved page, if it cannot find any disease in *S1* in that page but some disease in *S2* is found, then classify it as a "negative" example. (The underlying assumption is that if an unknown side effect is caused by d, then d also causes at least one known side-effect. Thus, if no known side effect is obtained, we assume that d does not cause any side effect.)

(4) Submit a query of the form <d, side effect, s1> to PubMed or Google, where s1 is a known side effect in *S1* of drug d.

(5) For each retrieved page, if some disease s1 in *S1* is found, and it cannot find words such as "safe" or "not" in the vicinity of s1, then classify it as a "positive" page ( That is, a side effect of the drug is found in the page.)

(6) Obtain 50 classified "positive" pages and 50 classified "negative" pages. Manually identify which ones are classified correctly. Compute the precisions.

## 4.2 Experimental Results

The following sets of experimental results are shown below.

(1) The precision of our system. This is given by two "precision" values. The first precision value is the number of pages retrieved by our system which describe actual side effects of the drug as judged manually/ the number of pages which are retrieved by our system. This will be referred to as precision-accept. Since this requires manual operation, we can afford to do this for 5 drugs only, with Google retrieving 100 pages for each drug. The second precision number is the number of pages rejected by our system (among the 100 pages) which do not describe any side effect of the drug/ the number of pages rejected by our system. This will be referred to precision-reject. Ideally, if our system is perfect, both precision-accept and precision-reject should be 1. The results for the 5 drugs are given in Table 1.

**Table 1. Accept and Reject Precisions**

| Drug Name | Precision-accept | Precision-reject |
|---|---|---|
| **Caduet** | 80% | 93.8% |
| **Vioxx** | 94.4% | 86.6% |
| **Lipitor** | 77% | 92% |
| **Synthroid** | 100% | 87.7% |
| **Avastin** | 100% | 77.3% |
| **Average** | 90.3% | 87.5% |

On the average, only 16.4 pages out of 100 pages are selected by our system. Thus, based on the results given in Table 1, the accuracy of accepting a relevant page by our system and that of rejecting an irrelevant page are high, in spite of the fact that most pages retrieved by Google are irrelevant.

Another experiment we perform is as follows. We submit queries of the form < drug name, side effect > to Google ( instead of dropping the phrase "side effect") and for each drug, we retrieve the top 17 pages ( versus 16.4 pages our system accepts on the average for each drug.) The precision for each of the 5 drugs is shown in Table 2. The average precision by Google is 61.2% versus 90.3% by our method.

**Table 2. Precisions for top 17 pages for Google**

| Drug Name | Precision |
|---|---|
| **Avastin** | 52.9% |
| **Cadnet** | 76.5% |
| **Lipitor** | 58.8% |
| **Synthroid** | 47.1% |
| **Vioxx** | 70.6% |
| **Average** | 61.2% |

(2) The side-effects of the following drugs which are retrieved by our system, but they are not recognized by FDA are given in Table 3.

**Table 3: Extracted Unrecognized Side Effects**

| Drug Name | Unrecognized Side Effects |
|---|---|

| | |
|---|---|
| **Asendin** | Breast cancer |
| **Paxil** | Breast cancer |
| **Anafranil** | Breast cancer |
| **Norpramin** | Breast cancer |
| **Surmontil** | Breast cancer |
| **Rhotrimine** | Breast cancer |
| **Prilosec** | Pneumonia |
| **Betaseron** | Dehydration, Hemorrhage |
| **Kaletra** | Inflammatory oedema of the legs |
| **Accutane** | Watery eye |
| **Vioxx** | Clot<br>Heart attack, Stroke |
| **Meridia** | Increased sex drive<br>Inflammation |
| **Sanctura** | Tremor, Seizures |
| **Norvasc** | Nosebleed<br>Nasal inflammation |
| **Boniva** | Influenza, Constipation |
| **Uroxatral** | Swelling of ankles and legs |
| | Yellowing of skin or eyes |
| **Omacor** | Nausea |

(3) The accuracy of our techniques to automatically generate positive and negative examples, which can be used as training data are reported. We use the algorithm given in Section 4.1 to generate 50 positive examples and 50 negative examples. The accuracy of the 50 positive examples is 98% i.e. among the 50 examples which our system classifies as positive examples, there is one error only. The accuracy of the 50 negative examples is 96%.

## 4.3 Validation

The side effects of the drugs given in Table 3 are examined by Patricia M. West, a licensed pharmacist and drug information specialist. Various sources, including the following sources [E01, B05, M05, Mc05, T05], are used for the validation of the side effects. The side effects of the following drugs which are not recognized by FDA are confirmed by her.

Prilosec: Pneumonia
Accutane: Watery eye
Uroxatral: Yellowing of skin or eyes

It should be noted that her validation of the side effects is conservative. For example, there was a study which indicates Asendin, Paxil, Anafranil, Norpramin, Surmontil and Rhotrimine may double the risk of breast cancer, but more recent data with larger populations have not found an increased risk. In this case, the side effects of these drugs are not validated. Any side effect discovered by the system but which can not be confirmed by any source available to the expert is assumed to be not validated. Furthermore, side effects obtained by our system which are closely related to known side effects are also assumed to be not validated. For example, nausea has been identified by our system to be a side effect of Omacor, but since vomiting is a known side effect of the drug, nausea is not an unrecognized side effect. The side effects of Vioxx are so well publicized that we do not consider them to be unrecognized.

# 5. CONCLUSION

In this paper, we provide a methodology to mine unrecognized side effects of drugs. Our experiments show that the obtained results involving Vioxx are consistent with the recent reported news, namely they cause heart problems. We have also identified officially unrecognized side effects of several other drugs.

Our methodology consists of capturing the relationships of features of a domain (in the drug domain, medical terms, diseases/symptoms, distances between certain content words are the features) with the subject of interest (side effect) by submitting a few queries and determining the relevance of the retrieved pages. Since the determination of the relevance of the retrieved pages by humans can be time consuming, we propose an automatic process to generate the positive and the negative examples and show that the proposed process is highly accurate. In this paper, we utilize a neural network to capture the desired relationships. However, other machine learning techniques such as [LSCP96, Mit97, YL99, HaMa00] can be utilized. Based on the captured relationships, queries of the same domain but involving other entities (drugs) can be processed with high precision. Since some data from the Web are likely to contain errors, we eliminate the noise by discarding data involving low frequencies of occurrences. In order to present an overview of the results to the user, only aggregate data (data involving total frequencies of occurrences) and the "most important" passage of each accepted page are presented. The side effects of three drugs which are not recognized by FDA are validated by an expert. Actually our system has identified quite a few additional unrecognized side effects of other drugs. However, the validation process is time consuming and these unrecognized side effects have not been validated by our expert, due to the lack of time.

Traditionally, an information system or a search engine retrieves documents based on a given query. As we all

know, a lot of irrelevant documents are retrieved, in spite of various advances [Kwok03, LLYM04, RW99, KuLe04]. A key reason for the low precision is that relationships among content words in documents are not captured precisely. In contrast, our methodology of retrieval is a two step process. In the first step, our system attempts to capture relationships among content words by applying a machine learning algorithm to a set of training examples. In the second step, actual retrieval takes places by utilizing the captured relationships among the content words. We note that the captured relationships are likely to be domain specific and probably problem specific. In other words, for each type of problems or each domain type, the system has to be trained, since the semantic information conveyed by the relationships among content words may differ from one type of problems to another. Furthermore, the vocabulary used in one domain may differ from that in a different domain.

In spite of the restrictions mentioned in the last passage, we believe our methodology to achieve "accurate retrieval" is applicable to a large variety of problems. For example, if we are interested in finding the complications caused by medical procedures, we proceed in the same way as we have done for finding the side effects of drugs. Specifically, we train a classifier from the training data of a set of medical procedures and then we apply Web search and the classifier to other medical procedures. As another example, suppose we are interested in finding the distance between any two planets. Then we can train the system for some specific pairs of planets, say (Jupiter, Mars) and apply the trained system to other pairs. Our plan is to demonstrate that this methodology can be applied successfully to a wide variety of real life problems. If this is successful, this may form a basis for highly effective retrieval, in which queries are classified into different types (possibly millions of types) and for each query type, some training is performed for some queries and then the trained system is applied to other queries of the same type.

## 6. REFERENCES


[AH03] S. Ahmad. Adverse drug event monitoring at the food and drug administration. J Gen Intern Med 2003, 18:57-60.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval. Addison-Wesley, Wokingham, UK, 1999.

[B05] Burnham TH, editor. Drug facts and comparisons. St. Louis: Facts and Comparisons; 2005

[Couz04] J. Couzin. Drug safety. Withdrawal of Vioxx casts a shadow over COX-2 inhibitors, Science, Oct 2004.

[E01] *Eye 2001;15(Pt 1):115-6.*

[Harm95] D. Harman TREC –4, NIST, 1995.

[Harm96] D. Harman, TREC-5, NIST 1996.

[HBL98] B. Honigman, D. W. Bates and P. Light, Computerized Data Mining for Adverse Drug Events in an outpatient Setting, Proceedings of the 1998 AMIA Annual Symposium.

[LSCP96] D. Lewis, Robert E. Schapire, James P. Callan, Ron Papka. Training Algorithms for Linear Text Classifier. SIGIR 1996.

[GK98] D. Graupe and H. Kordylewski. A Large Memory Storage and Retrieval Neural Network for Adaptive Retrieval and Diagnosis. Internat. J. Software Eng. and Knowledge Eng, 1998.

[HaKa00] J. Han and M. Kamber, Data Mining: Concepts and techniques, Morgan Kaufmann, 2000.

[Ko82] T. Kohonen, Self-organizing formation of topologically correct feature maps, Biological Cybernetics, 43(1):59-69, 1982

[Kwok03] K. Kwok, L. Grunfeld, N. Dinstl, and P. Deng TREC 2003 Robust, HARD and QA Track experiments using PIRCS, TREC 2003.

[Kule04] O. Kurland and L. Lee Corpus Structure, Language Models and Ad Hoc Information Retrieval, ACM SIGIR, 2004.

[LAWH02] K. Lasser, P. Allen, S. Woolhandler, D. Himmelstein, M. Wolfe and D. Bor. Timing of new black box warnings and withdrawals for prescription medications. JAMA 2002.

[LLYM04] S. Liu, F. Liu, C. Yu and W. Meng An effective approach to document retrieval using Wordnet and recognizing phrases ACM SIGIR Conference, pp.266-272, 2004.

[Mc05] McEvoy G. American Hospital Formulary Service 2005. Betheseda: American Society of Health-System Pharmacists.

[M05] Micromedex® Healthcare Series, (electronic version). Thomson Micromedex, Greenwood Village, Colorado, USA. Available at: http://www.thomsonhc.com (cited: 05/24/2005)

[NMTM99] K. Nigam, A. Mccallum, S. Thrun, and T. Mitchell. "Text Classification from Labeled and Unlabeled Documents using EM", Machine Learning 1999.

[PBLLOE02] Eugene P. van Puijenbroek, Andrew Bate, Hubert G. M. Leufkens, Marie Lindquist, Roland Orre5 and Antoine C. G. Egberts, A comparison of measures of disproportionality for signal detection in spontaneous reporting systems for adverse drug reactions, pharmacoepidemiology and drug safety 2002.

[RW99] S. Robertson and S. Walker Okapi. at TREC-8, Trec-8, 1999.

[SALT89] G. Salton. Automatic Text Processing, Addison Wesley, 1989.

[T05] Thomson MICROMEDEX. Drug information for the health care professional. 25th ed. Greenwood Village: Thomson MICROMEDEX; 2005.

[Voor01-04] E. Voorhees edited, Question Answering Track in TREC, 2001, 2002, 2003, 2004.

[YL99] Yiming Yang and Xin Liu. A re-examination of text categorization methods. ACM SIGIR, 1999.

[Mit97] T. Mitchell. Machine Learning. McGraw-Hill, 1997

[Van79] C. J van Rijsbergen. Information Retrieval Butterworth 1979.

[YuMi88] C. Yu and H. Mizuno "Two learning algorithms in information retrieval", ACM SIGIR 1998.