DEVELOPING COMPUTER-GENERATED PUBMED QUERIES FOR IDENTIFYING DRUG-DRUG INTERACTION CONTENT IN MEDLINE

By

Stephany Norah Duda

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Biomedical Informatics

December, 2005

Nashville, Tennessee

Approved:

Professor Kevin B. Johnson

Professor Constantin F. Aliferis

Professor Randolph A. Miller

ACKNOWLEDGEMENTS

I would like to thank the National Library of Medicine, whose generous support has made this research possible (Training Grant T15 LM 007450-03). I also want to extend heartfelt thanks to Christine Sommer, George Robinson, and Jerome Osheroff for their insights into drug database development and maintenance, and to Patricia Lee for her work building PubMed queries for this project.

The three members of my thesis committee have been invaluable in the preparation of this work. I would like to thank Dr. Constantin Aliferis for making numerous contributions to the design of these experiments and for providing my research with a sense of structure. His thorough understanding of machine learning theory and application has given me an excellent introduction to the field. I will always remember that even in my moments of pre-presentation panic he believed in the validity of this research topic, and thereby instilled in me a confidence in the potential of this work.

I also want to thank Dr. Randolph Miller, who always managed to find time for my research in the midst of his demanding projects. It has been an honor to have a renowned informatics pioneer read every draft I put on his desk multiple times and contribute such constructive and insightful comments. His encouragement has made me strive to be a more discerning scientist and a better writer.

I especially wish to thank Dr. Kevin Johnson, my committee chair, for his insight and guidance in completing this research. He has inspired me with his passion for research and taught me more useful skills than I can enumerate. Indeed, if any

presentation of my research has been interesting, it is solely thanks to his input and instruction. Dr. Johnson has been a brilliant and considerate mentor who trusted in my work and abilities, even when I didn't think I was up to the task. Our research meetings (occasionally quite off-topic) have been some of my most educational and enjoyable times in graduate school. I truly look forward to continuing this work.

I would like to thank all the members of the Department of Biomedical Informatics, especially my fellow graduate students and Ms. Rischelle Jenkins, the heart of our informatics family. Her optimism and moral support have cheered me up on numerous occasions. I would also like to thank Yin Aphinyanaphongs for many hours of valuable research discussion and Alexander Statnikov for running experiments and being incredibly generous with his time and help. Their detailed explanations (and patience in giving them) have given me an understanding of machine learning methods that I could never have hoped to achieve on my own. I particularly want to thank Randy Carnevale and Laura Brown, not only for their help with MATLAB, PHP, and SQL, but also for their constant encouragement and kindness, and for regularly switching on the tea kettle for me on late research nights. I could not have asked for better friends.

Finally, I would like to thank my parents, Hilary and Francis, and my brother Brendan. They have helped me get past every milestone, celebrated my successes, and consoled me during setbacks. I have made it this far because of their constant love and support.

Thank you for everything.

TABLE OF CONTENTS

Page

Appendix

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ADE ..... adverse drug event

API ..... application programming interface

AUC ..... area under the receiver operating characteristic curve

CPOE ..... computerized physician order entry

CUI ..... Concept Unique Identifier

DDI ..... drug-drug interaction

DDI+ ..... positive drug-drug interaction article

DDI- ..... negative drug-drug interaction article

DDII ..... Drug-Drug Interaction Initiative

DT ..... decision tree

FDA ..... U.S. Food and Drug Administration

FPR ..... false-positive ratio

HITON ..... a Markov Blanket induction and variable selection algorithm

HITON-MB ..... HITON, Markov Blanket variant

HITON-PC ..... HITON, Parents and Children variant

KB ..... knowledge base

MeSH ..... Medical Subject Headings

MMTx ..... MetaMap Transfer

NIH ..... National Institutes of Health

NLM ..... National Library of Medicine

NLP ..... natural language processing

NPV ..... negative predictive value

PMID ..... PubMed Unique Identifier

PPV ..... positive predictive value

ROC ..... receiver operating characteristic

SVM ..... Support Vector Machine

TPR ..... true-positive ratio

UMLS ..... Unified Medical Language System

VA ..... Veterans Administration

INTRODUCTION

Concurrent administration of "interacting" medications causes patients to experience unexpected physiologic effects and alterations in metabolic pathways. Such drug-drug interactions occur frequently, and can have expensive and dangerous consequences. Unfortunately, existing computerized alerting systems that are designed to prevent such hazardous medication errors often fail to impact clinical decision-making. One fault stems from the unreliability of knowledge bases providing drug interaction content for these alerting systems. Drug-related information in such systems is often outdated, clinically insignificant, or even incorrect. By improving the coverage and accuracy of the drug-drug interaction information in available databases, it may be possible to improve delivery of safe and cost-effective patient care.

Identifying new drug-drug interaction (DDI) content in the literature, however, is an expensive and manually intensive process. The current project investigated the feasibility of automating portions of the DDI identification process. This report details the first step in this series, and discusses improving the yield from the National Library of Medicine's MEDLINE database, a potential source of drug interaction information. In particular, this study evaluated computer-generated Boolean queries as an alternative to manually constructed Boolean queries for extracting drug-drug interaction information from MEDLINE.

This research focused on improving the usefulness of a single source of DDI content. Later work should explore other DDI information resources and develop tools

for extracting interacting drug pairs from text, evaluating the type and severity of the interaction, and applying this knowledge to drug database creation and maintenance.

Thesis Contents

This report is divided into five chapters describing the different stages of this research project. Chapter I presents an overview of the alerting systems designed to prevent drug-drug interactions and the reasons these systems often fail. It introduces MEDLINE as a possible source of DDI knowledge, describes how MEDLINE's value might be improved through better search techniques, and reviews published methods of information retrieval. The design of computer-generated Boolean queries is also discussed.

Chapter II describes the process of developing a machine learning classifier that can distinguish between "positive" and "negative" examples of DDI citations in MEDLINE. Chapter III confronts the complexities of implementing such a computerized model and describes experiments that decomposed the machine learning classifier into a traditional Boolean query. The Boolean queries' performance is presented in Chapter IV. Finally, Chapter V summarizes the research and discusses applications and future directions for this work.

CHAPTER I

OVERVIEW OF DRUG-DRUG INTERACTION PREVENTION AND DISCOVERY

Unwanted drug-drug interactions pose a serious threat to patients and present the medical field with costly and frequent medication-related illnesses. The first part of this chapter presents the impact and severity of adverse drug events and drug-drug interactions. The subsequent sections summarize the benefits and the drawbacks of computer-based systems designed to reduce the occurrence of DDIs and explain how the content of these systems might be improved through effective use of MEDLINE as a source of high-quality DDI information. The later sections of this chapter discuss techniques of document classification using MEDLINE records and illustrate the use of support vector machines as classifiers. The final part of this chapter details the research problem investigated here and outlines the process of locating drug-drug interaction content in MEDLINE.

Severity of Adverse Drug Events and Drug-Drug Interactions

Adverse drug events – defined as patient injuries resulting from drug-related medical treatment [1] – are a serious concern of both clinicians and consumers in today's pharmacologically complex healthcare environment. The widespread use of medications – 3 billion prescriptions in 2000, almost double the number a decade previously [2] – suggests a sizable percentage of the population is at risk for adverse drug events. In the United States, adverse drug events (ADEs) have been reported to affect 6% of

hospitalized inpatients [3] and up to 26% of adult outpatients in primary care [4]. Moreover, ADEs are purported to be responsible for 76,000 fatalities in the United States each year, making these medical disasters the sixth leading cause of death among adult Americans [5].

Although not all ADEs result in injury, serious ADEs produce higher hospital admission rates, longer hospital stays, lost worker productivity, and lower patient satisfaction [6]. In hospitalized patients alone, such ADEs are associated with a two-fold increased risk of death [7]. The most serious ADEs can be life-threatening and cause irreversible harm, particularly to the very old and very young: nursing home patients are more likely than middle-aged adults to suffer serious injury or death as a result of ADEs, and among the general population, one third of patients permanently disabled by ADEs are under 10 years old [8-11].

The financial burden of ADEs is also significant; the Institute of Medicine's ground-breaking *To Err is Human* report estimates preventable ADEs affecting inpatients cost hospitals $2 billion a year [1]. When extended to include outpatients, the increased illness and death associated with ADEs may carry a yearly cost of over $70 billion [12]. Most important, approximately 30% of these ADEs are preventable [3, 8, 13, 14].

Drug-drug interactions (DDIs) make up a sizable subset of preventable ADEs in both the in- and outpatient population [10, 15]. DDIs occur when the physiologic or side effects of one drug are altered by the presence of another drug in the body, producing changes that are often unwanted and can be harmful [16]. Drug-drug interactions can trigger increases in drug toxicity, changes in drug efficacy, and treatment failure. Among primary care patients, 58% are concerned about DDIs [17], and drug interactions in this

population are an important cause of emergency department visits [18]. Studies estimate that as many as 8% of patients experience some DDI during a period of medication use [19], and that these interactions are responsible for 2.8% of all hospitalizations in older populations [20]. Indeed, the incidence of drug-drug interactions in patients may range from 4.7% to 11.1% [21], costing the healthcare system close to 1.3 billion dollars each year [22].

Preventing Drug-Drug Interactions

Efforts to increase clinician awareness about drug interactions are hampered by the sheer number of new drugs and potentially serious drug combinations. There are currently 20,000 FDA approved prescription drugs marketed in the U.S. and the FDA approves approximately 340 new and potentially interacting drugs every year [23]. Over 2,000 DDIs have been reported in the literature, some of which are based on a class of medications interacting with a specific ingredient, meaning a single DDI might affect hundreds of drug combinations [24]. The pace of discovery means many clinicians are unable to keep abreast of the latest pharmaceutical developments, including drug-drug interaction updates. A survey of 263 doctors practicing in the Southern California Department of Veterans Affairs system found that physicians were unable to identify 50% of contraindicated drug pairs [25]. On the other hand, doctors often do not differentiate between the properties of individual drugs and their corresponding drug classes, and therefore may assume certain drugs interact when they do not [26]. Given these considerations, 89% of physicians consider drug-drug interactions a risk in prescribing [26].

Often the responsibility for checking for DDIs is delegated to pharmacists, who already find themselves overwhelmed with prescriptions [27]. Yet many pharmacists may not be more informed than the prescribing doctors: half of all pharmacists dispense potentially lethal drug combinations without written or verbal warning to the customer [28].

Improving clinicians' awareness and monitoring of drug-drug interactions can improve delivery of safe and cost-effective patient care [29]. Computerized warning systems provide an effective solution: electronic prescribing tools coupled with drug alerting systems have been shown to change physicians' practices [30] and help decrease the incidence of DDIs [31]. Similar DDI warning systems can be installed in hospitals, clinics, and pharmacies to help protect patients against harmful drug interactions.

Drug knowledge bases, such as those developed and maintained by "knowledge vendors" like Micromedex, Medi-span, and First Databank (FDB), provide the drug content for DDI alerting systems. These knowledge bases (KBs) are repositories for pharmaceutical knowledge represented in machine-processable form. When integrated with applications such as physician order entry systems, electronic prescription writing tools, and pharmacy dispensary systems, these drug-drug interaction modules can generate electronic alerts for conflicting drug regimens.

The current process of building DDI knowledge bases incorporates expert knowledge and information from the manual review of pharmacy information bulletins (e.g. PharmacyOneSource.com), drug company publications, FDA warnings, and the newsletters of professional associations like the American Society of Health-System Pharmacists [32]. DDI KB developers also review table-of-contents alerts and articles

from high-impact journals, though the number of journals they can cover is limited and the issues are not always up to date[33, 34]. Citation repositories such as OVID and NLM's MEDLINE are often used as secondary reference sources [32]. Any new drug-drug interaction information is reviewed by panels of clinical pharmacists, who assemble DDI monographs for each interaction. These monographs contain information on the offending drug pair, the interaction severity, patient risk factors, and treatment options [35].

Drawbacks of Drug Interaction Alerting Systems

Despite great concern over the impact of drug interactions, many knowledge bases fail to include important drug interactions and contain outdated, irrelevant, or even incorrect information [36]. One evaluation of six computerized DDI screening programs reported that only two programs could detect all serious interactions, and even so, the information they presented was weak and unhelpful in treating patients [37]. A study of 9 pharmacy drug systems by Hazlet and colleagues found that DDI software systems were unable to detect clinically significant drug interactions one third of the time[38]. A similar hospital system failed to provide warnings for 70% of organ transplant-related drug interactions deemed dangerous by a panel of experts [39]. When ported to handheld devices, most of these drug interaction alerting programs display similarly poor performance [40, 41].

The problem of establishing "truth" related to DDIs is significant; even the most commonly used drug-drug interaction compendia, in both printed and electronic format, have been shown to contain serious discrepancies in their listing and severity rating of

DDIs [36]. An updated study of four popular DDI knowledge sources revealed that of 406 major, clinically relevant drug-drug interactions, only 9 were listed in all four compendia. Indeed, 72% of these highly important DDIs were listed in only one source [42]. These gaps in DDI KB coverage have resulted in pharmacists accidentally prescribing dangerous drug combinations even though drug interaction alerting software was in place [43].

Clinicians insist that excessive clinically insignificant DDI alerts pose an additional problem; not only does the annoyance discourage use of the system, the "noise" generated by a deluge of alerts leads many physicians to ignore most interaction warnings as irrelevant or trivial [25]. This "alert fatigue" is increasingly reported in the literature. In a 2003 study by Weingart and colleagues, physicians in five Boston adult primary care clinics felt that as little as one third of the alerts they received were appropriate to the situation [44]. A more recent study found this number to be as low as 11% [45]. Despite the usefulness of some drug warnings, the doctors studied in Boston overrode 89.4% of their computerized order system's "high-severity" DDI alerts [44]. Two similar studies, one of 42 community pharmacies, the other of orders at a large VA hospital, found that both pharmacists and doctors overrode up to 88% of drug interaction alerts [46, 47].

Other concerns arise when DDI systems treat all members of a drug class the same, neglect to include patient risk factors, and offer no drug alternatives or recommendations about managing patients with ongoing drug interaction incidents [48, 49]. Many drug interaction databases are also not updated frequently enough to reflect the latest DDI developments [50].

Poor quality DDI information can harm patients, increase expenses, and produce medical practitioners who have widely differing opinions on the clinical relevance of many drug-drug interactions [45, 46]. Many institutions find that they cannot use DDI warning software off-the-shelf without major adjustments. In order to create effective tools, large medical centers and commercial pharmacy chains must customize alerting thresholds and focus the content of the DDI knowledge bases supporting these systems. At one U.S. hospital, a panel of DDI specialists evaluated 56 interactions rated by the vendor of the system as "high significance" and were able to reduce this number to 28 clinically significant interactions [51]. A second hospital was forced to implemented its own "safety net" system to catch dangerous interactions missed by commercial software [52].

Clinicians note that drug information and database vendors need to do a better job of ensuring the drug-drug interactions they list are clinically important [53]. These vendors' products often include and generate alerts for every known or suspected DDI, possibly motivated by medical and legal concerns [54]. Indeed, one major DDI knowledge base lists over 100,000 potential drug interactions for fewer than 20,000 FDA-approved drugs [23]. The inclusivity of vendor systems has become such a concern that the US Pharmacopeia recently introduced a strategy to counteract this expansion. This plan, known as the Drug-Drug Interaction Initiative (DDII), aims to address the problem of outdated, redundant, and clinically insignificant drug alerts by establishing an industry standard for rating drug interactions [53].

Improving Drug-Drug Interaction Knowledge Bases

As emphasized by the DDII, improving the content of drug-drug interaction knowledge bases is the first step to improving the quality of DDI alerts [27]. The incomplete and unreliable information in current DDI KBs is in part the result of inadequate techniques for collecting, filtering, and maintaining drug interaction knowledge. Improved DDI information retrieval techniques may therefore assist in the development of more reliable knowledge bases.

This problem of identifying relevant DDI information, however, is complicated by the overwhelming amount of new drug publications. The FDA approves approximately 340 new medications each year, each of which generates multiple drug company publications, pharmacy alerts, FDA guidelines, and journal articles [23]. Indeed, current pharmaceutical research alone results in approximately 300,000 MEDLINE citations per year that are labeled with MeSH terms from the "Chemicals and Drugs" category, and a representative reference publication, *Physician's Desk Reference*, has grown from 2,787 pages in 1995 to 3,440 pages in 2005.

Unfortunately, filtering this published material for new drug information remains predominantly a manual task, which suggests that the clinical pharmacists responsible for maintaining DDI knowledge bases cannot easily review every published article [34]. Consequently, both new and revised drug interaction data may be overlooked.

MEDLINE as a Source of Drug Interaction Information

The National Library of Medicine's MEDLINE database is a major repository of biomedical literature references from nearly 4800 U.S. and international journals [55] and

provides a rich source of high-quality DDI information [56]. In addition, studies in other fields suggest that MEDLINE searches may reveal high-quality content that is not found by consulting experts or other databases [57].

The NLM (National Library of Medicine) provides free access to MEDLINE through a search portal at www.ncbi.nlm.nih.gov/pubmed/ (PubMed). PubMed's web interface allows users to run Boolean queries against the MEDLINE database and returns an organized set of hyperlinked results. The major search fields used for domain queries (in contrast to searching for a particular paper) include title, abstract, and the Medical Subject Headings (MeSH) that have been hand-selected by MEDLINE indexers based on a paper's content [58]. Clinicians often search MEDLINE using simple search strategies composed of text words and MeSH terms and many are familiar with PubMed-formatted Boolean queries [59, 60].

Unfortunately, MEDLINE is often overlooked by drug database developers because searching the MEDLINE database for DDIs and sorting the results is a manually intensive, and therefore expensive, task. Yearly changes in MEDLINE's MeSH vocabulary result in inconsistent indexing, which complicates both document identification and query maintenance [61]. In addition to problems with available search tools, studies have shown that many naïve end-user searches only retrieve one fourth of relevant articles [62] and that even expert searchers capture only a small fraction of the relevant literature with their targeted queries [63, 64].

These insights have motivated the development of information retrieval methodologies for locating relevant documents in large bibliographic databases and text corpora. Early methods focused on improving the return of MEDLINE queries by

applying search filters for high-quality clinical information [65-67]. These strategies, pioneered by Haynes and colleagues, used search terms submitted by experts to design Boolean queries for content filtering. These queries were subsequently tested against a corpus of hand-labeled citations. Several recent studies have employed the same methodology to identify MEDLINE articles about health services research, sleep studies, obstetrics, and randomized controlled trials for Cochrane review [68-71]. Word frequency analyses and statistical measures have also been used to develop queries that identify MEDLINE articles about dental research, stem cells, and diagnosis. [72-74].

Similar studies applied these techniques in conjunction with the NLM's UMLS (Unified Medical Language System) Metathesaurus to identify molecular binding terminology and key clinical concepts [75-77]. The UMLS maps the terms of its diverse source vocabularies to unique concept identifiers (CUIs), clusters equivalent headings, and provides inter-concept relationships within and between over 100 biomedical vocabularies [78]. Bodenreider's research on the application of UMLS to condition and disease categories suggests that the classifications of the UMLS Metathesaurus are representative of a document's content and that the concept-level relationships in the UMLS Metathesaurus might prove useful for document classification [79].

Statistical and natural language processing (NLP) techniques for identifying relevant MEDLINE information have been particularly popular in bioinformatics, where they have been used to discover inhibitory drug-drug interactions, protein-protein interactions, and citations with pharmacogenetic knowledge [80-82]. Similar work has focused on using natural language processing (NLP) techniques to extract biomedical information and ADE reports from other sources of biomedical text [83, 84].

Although old, manually generated queries can still perform admirably over a decade later [85], new research suggests that manual and simple statistical information retrieval methods can be improved upon by machine learning classification techniques [86]. Indeed, computers are capable of learning text patterns that identify specific articles and can often continue to classify unseen examples with good performance. Using a computer-generated query may be less costly and produce results that are better than, or at least comparable to, current expert-guided methods [87, 88].

Support Vector Machines for Text Classification

One of the most successful automated "machine learning" methods is support vector machine classification [89]. Support vector machines (SVMs) are one of many supervised learning techniques, a subset of machine learning techniques that can be used to create output-prediction functions given a set of labeled training data. When used for binary classification, SVMs project vectors of variables into a higher feature space and identify the maximum-margin hyperplane that separates the positive and negative training examples.

Figure 1 provides a simple example of a linear SVM classifier. In this 2-dimensional problem, the data are linearly separable, but although there are many lines that can definitively separate the circles from the squares, as shown in (a) on the left, only one is a maximum-margin plane (i.e. it maximizes the distance between itself and the nearest square and circle.) Figure 1 (b) depicts the optimal linear separator (bold line). The datapoints in dark red and green are the support vectors, which define the optimal margin (shown with dotted lines).

Linear Support Vector Machine

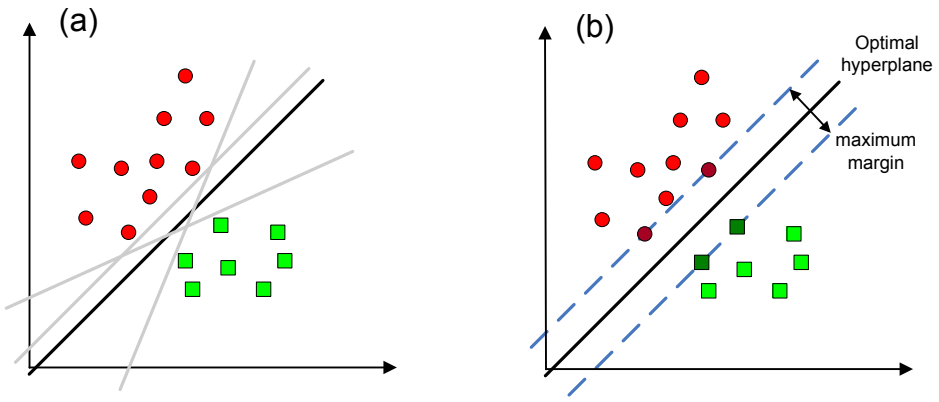

Figure 1: Example of a Linear SVM.
This example illustrates the most elementary type of SVM classifier: the linear SVM. The red dots and green squares represent two classes of data, which the SVM attempts to separate with a line (hyperplane). Figure 1(a) depicts four hyperplanes that can separate the two classes of data without error. The optimal hyperplane with the maximally separating margin is shown in Figure 1(b). The dark red circles and green squares represent the support vectors. These datapoints are closest to the optimal hyperplane and define the edge of the margin (blue dotted lines).

Some data are not linearly separable in the given input space, however, as in Figure 2(a), below. No straight line can be drawn to separate the red and green datapoints. But the data still might be separable using a more advanced function, such as the polynomial depicted in Figure 2(b). The SVM classifier applies a kernel function to transform the input space into a high-dimensional feature space. Figure 2(b) shows the hyperplane drawn by a polynomial kernel of degree 2, projected back into the input space.

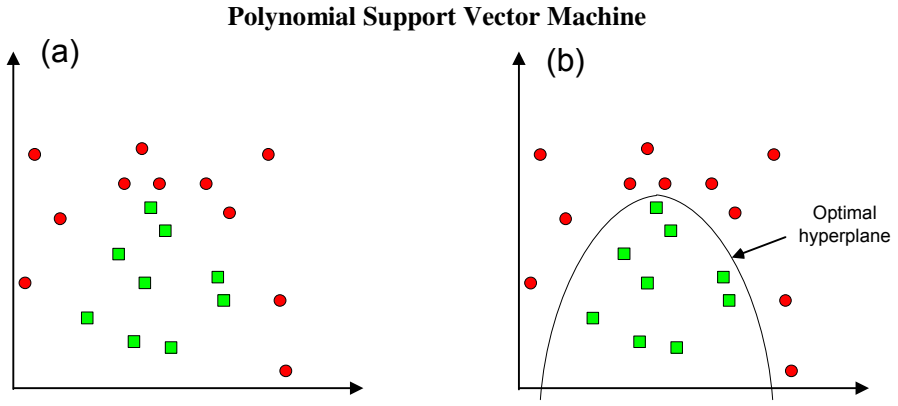

Figure 2: Example of a Polynomial SVM
The red circles and green squares in Figure 2(a) represent two classes of data that are not linearly separable. A non-linear SVM classifier can separate the data classes using a polynomial function. Figure 2(b) depicts the hyperplane identified by the polynomial SVM. When mapped back to the input space, this hyperplane appears parabolic.

Given enough extra dimensions, the SVM classifier can linearly separate the data, but the optimal hyperplane it defines may correspond to any number of strangely shaped, non-linear surfaces when mapped back to the original input space [89].

The SVM model also includes a cost parameter, *C*, to penalize misclassifications. Greater values of *C* produce stricter, though perhaps less generalizable SVM classifiers, since the focus is on avoiding misclassification rather than maximizing the margin [90]. Simple SVM models use a constant cost parameter.

Although SVMs are particularly suited to handling large feature spaces, they can suffer from overfitting, particularly when training examples are sparse or the model is trained for too many iterations. In this situation, the SVM grows overly complex and becomes tailored to random variances in the training data that have no effect on the target output. The SVM's performance on the training data is maintained, but its ability to generalize to unseen cases diminishes.

Cross-validation and feature selection are two common methods of avoiding overfitting [91]. In "train-test-validate" cross-validation, the data are partitioned into three, non-overlapping sets, as depicted in Figure 3. The first is the training set, which the learning algorithm uses to develop a candidate model. This model is tested on the validation set, and tuned until it performs well on both the train and validation sets.

Since the validation dataset is used in the model generation phase, it is not a good set on which to evaluate the model's ability to generalize to unseen data. In this case the training and validation sets are used iteratively in model development and selection, while the test set is put aside to evaluate the final model. The best final model, then, is the one that produces the lowest error over the test set [89].

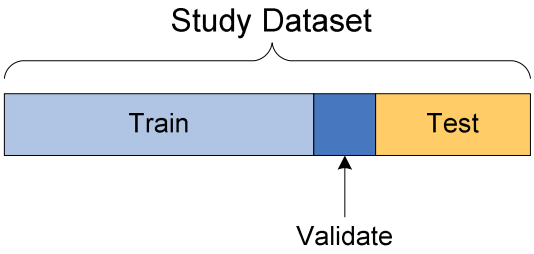

Figure 3: Partitioning the Study Dataset
The allocation of data samples to training, validation, and test sets is depicted in Figure 3. The training and validation sets (blue) are used to develop and optimize the classifier. The test set (yellow) is set aside and used to test the generalizability of the final, optimized model.

In a commonly-used extension of this method known as *k*-fold cross-validation, the procedure described above is repeated *k* times, each time using a different portion of the training set for validation. The data are partitioned into *k* disjunct sets and each set is used in turn as a validation set, while the remaining data are combined into the test set.

This method is particularly effective for preventing the overfitting that can accompany small datasets [92].

A second method of controlling overfitting, feature selection, aims to reduce the size of the dataset by keeping only those dataset variables which have high discriminatory power. Only this subset of features is used to train a classifier, which improves the speed and, potentially, the performance of the machine learning task [93].

Feature selection has also been shown to be a powerful tool for reducing the high dimensionality of the feature space, which is especially useful for text categorization datasets [94]. Large datasets of text words and phrases often contain many non-informative words that are only present in a single document. The SVM model is less likely to overfit the data once noisy variables are removed, since remaining features are associated with the target output. With these precautions, SVMs show excellent generalization capability in comparison to other machine learning techniques [95].

Overall, SVM models have proven to be more successful at classifying text than other machine learning techniques. The pioneering study by Joachims compared SVMs with alternative methods of text categorization, including the Rocchio algorithm popular in information retrieval, a distance-weighted k-nearest neighbor classifier, the C.45 decision tree/rule learner, and a Naïve Bayes classifier. On two separate corpora, the SVM classifier substantially and consistently outperformed the other four learning methods [96]. A recent study by Aphinyanaphongs and colleagues reported the successful application of SVM models to MEDLINE citations, identifying high-quality MEDLINE articles with greater sensitivity and specificity than Naïve Bayes classifiers or text-specific boosting [97].

Evaluating SVM Performance

Results of the SVM model are often presented in terms of the area under the receiver operating characteristic (ROC) curve. This curve, shown in red in the three figures below, represents the trade-off between the true-positive ratio (TPR) and the false-positive ratio (FPR) of a classifier (also sensitivity and 1-specificity).

The "random classifier" in Figure 4 depicts an ineffective model whose classification is no better than chance. For each point along the diagonal line, the true-positive and false-positive ratios are equal, producing a line with a slope of 1. In contrast, the "perfect classifier" captures all the true positives with no mistakes. Its true-positive ratio is always one; its false positive ratio is zero; and the resulting area under the curve (AUC) is one.

The center graph in Figure 4 shows how a realistic classifier performs: it is neither useless nor perfect. The performance of an SVM model, however, can be measured by how close it approximates the perfect classifier. The better performing SVM models will produce an AUC closer to 1.0.

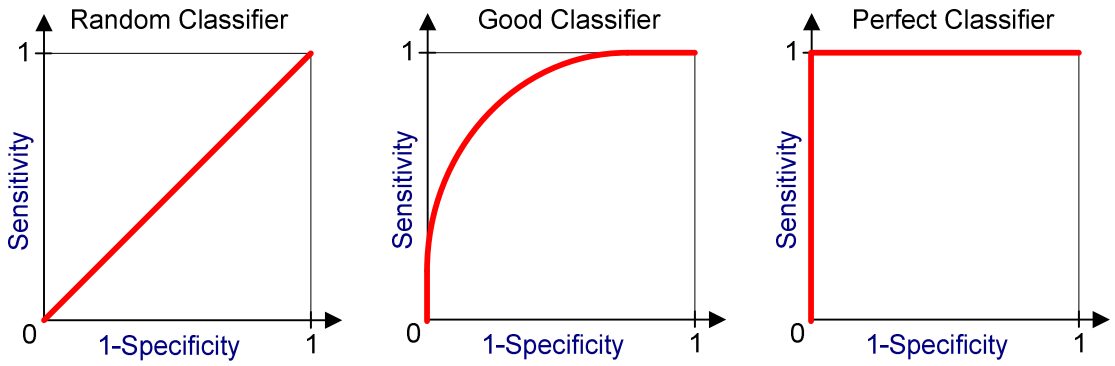

Figure 4: Sample ROC Curves
These three figures depict receiver operating characteristic curves (in red) that correspond to different classifiers. A random classifier (left) has equal true-positive (sensitivity) and false-positive (1-specificity) rates and therefore is a useless classifier. A classifier with good performance (center) has an ROC that curves above the x=y diagonal; its true-positive rate is greater than its false-positive rate. Perfect classification (right) results in a constant true-positive rate of 1. A classifier's performance can be represented by a measurement of the area under the red line.

This broadly applicable machine learning technique has already proven successful at text classification [98, 99] and identifying MEDLINE references according to their quality and usefulness in a clinical setting [97]. Text fragments such as words, word roots, and phrases have been used as input features.

Drawbacks of SVM Models

However, there are several limitations to using an SVM classifier, despite their overall good performance. Although an SVM model is not a true black box classifier, its internal architecture is considered to be non-intuitive [100]. This inherent complexity of SVM modeling makes it difficult to understand the basis for its results and thereby generate justifications for the model's predictions [101].

In addition, classifying text using an SVM model can be a time-intensive process, which makes it unsuitable for quickly locating new information in PubMed [102]. A

collection of unlabeled, candidate documents must first be assembled locally, and depending on the number of citations being categorized, it may take several minutes or hours to stem the text words of each citation, sum their occurrences per document, fit the results to the classifier's predetermined word-matrix, and run the classifier. To minimize the work involved, the text processing and document classification steps can be serialized in an application, though this type of classifier is not yet integrated with PubMed's query-based article retrieval.

Decision Trees and Queries

While poor transparency and a lack of PubMed integration make direct use of SVMs impractical, studies suggest that these models can be used as an intermediary step to produce a classifier whose decision-making logic is structured, transparent and interpretable. Decision trees are considered to be classification solutions of this type, since they can be represented as a series of if-then statements [92].

A decision tree, when used for text classification, is a predictive model that tests various attributes of a document and uses the results to sort the example into one of several predefined classes. A feature of the document is evaluated at each node in the tree structure, and the document is sorted down the appropriate path. The leaves of the tree represent possible classifications, and the branch leading to each leaf represents the conjunction of features that lead to a certain classification [89].

Decision trees can be designed to model any discrete output, including the output of SVM classifiers. A binary decision tree, however, has only two output classes: positive and negative. These decision trees can be used to generate targeted Boolean queries,

which can be used in query-based search engines such as the one provided by PubMed [103].

A binary decision tree's correct and incorrect classifications are used as a means of evaluating its performance. Correctly classified documents are true positives (TP) and true negatives (TN). Positive documents that have been classified as negatives are false negatives (FN) and incorrectly classified negatives are false positives (FP). These values are often listed in the 2x2 format shown in Table 1.

Table 1: Sample 2x2 Table
Binary classification results are most often presented in this classic 2x2 format, in which the classifier's predictions are compared to the documents' true classes (gold standard).

	Gold Standard		
Classifier Assignment	True (+)	False (-)	**Total**
True (+)	True positive (TP)	False positive (FP)	TP + FP
False (-)	False negative (FN)	True negative (TN)	FN + TN
Total	TP + FN	FP + TN	**N** = TP+FN+FP+TN

Four statistics calculated from these values are often used to describe the performance of a classifier: sensitivity, specificity, positive predictive value, and negative predictive value. The equations for these statistics are presented in Table 2.

Table 2: Calculating Measures of Performance

Statistic	Equation
Sensitivity	TP / (TP + FN)
Specificity	TN / (TN + FP)
Positive predictive value (PPV)	TP / (TP + FP)
Negative predictive value (NPV)	TN / (FN + TN)

The performance of Boolean queries can also be measured using these statistics. All documents returned by a query are considered positive according to the test; documents not returned are considered test-negative. When the numbers of positives and negatives in the document set are known, the 2x2 table can be filled in and the appropriate measures of performance can be calculated.

The terms 'recall' and 'precision' are equivalent to 'sensitivity' and 'positive predictive value,' respectively. Recall/precision is common measure of performance in information retrieval tasks. In machine learning, however, the same concepts are more frequently called sensitivity and PPV. The researchers have chosen to use the machine learning nomenclature in the remainder of this document in order to maintain continuity, since SVM performance is reported in these terms.

Study Overview

This study explored the application of manual and computer-based information retrieval methods to the drug-drug interaction domain. Specifically, the basic hypothesis was that text processing and machine learning techniques could identify a set of DDI articles more readily than manually created queries, and that the computer-generated

SVM models could successfully be decomposed into Boolean queries that rival manually generated queries at retrieving drug-drug interaction citations from MEDLINE.

As previously discussed, drug-drug interactions can seriously endanger patient health and are a source of financial concern to both hospitals and consumers. DDI alerting programs do a poor job of preventing unwanted interactions, in part because of the unreliable knowledge bases that power these systems. Institutions and individuals that manage drug-drug interaction databases, however, depend on complete and up-to-date information in order to create and maintain effective warning systems. This research had as a goal to advance MEDLINE's value as a source of DDI information in order to improve the availability and accessibility of high-quality DDI content, and ultimately assist in constructing more complete and relevant drug-drug interaction databases.

CHAPTER II

DEVELOPING A CLASSIFIER

Introduction

The literature presented in Chapter I documents the need for more complete, reliable drug-drug interaction knowledge bases. The project hypothesized that MEDLINE might provide a valuable additional source of drug interaction information for these knowledge bases. For this to occur, relevant DDI articles must be identified in an effective and efficient manner. Research indicates that automated document classification is a potentially useful method for pinpointing textual information in such MEDLINE citations. In particular, Support Vector Machines (SVMs) are a machine learning technique that has notably good promise for text classification in many domains. The project team proposed to investigate whether these computer techniques could be used to develop Boolean queries that identify DDI articles in MEDLINE better than manually generated Boolean queries.

The first part of the current study, described in this chapter, evaluated the ability of an SVM classifier to identify drug-drug interaction content in a limited corpus of MEDLINE citations. The baseline performance level was set by two expert Boolean queries, assembled by medical librarians specializing in MEDLINE searches.

The Methods section presents the study's definition of DDI information and describes how the project assembled a corpus of positive and negative DDI citations from MEDLINE, processed these records, and developed a dataset of the stemmed text words

and MeSH terms associated with each citation. Two-thirds of this dataset was used to train a series of SVM classifiers, which were then compared to the two expert, manually-generated queries using the remaining third of the data. The sensitivities and specificities of these classifiers are presented in the results.

The final section of Chapter II discusses the limitations associated with the size of the dataset, its classification of positives and negatives, and the use of stemmed text words and MeSH terms. It also presents the strengths and weaknesses of the SVM approach, and touches upon the motivation for the subsequent experiments presented in Chapters III and IV.

Methods

Defining Drug-Drug Interaction Content

The overall study objective was to locate MEDLINE citations that provide drug-drug interaction information potentially worth including in a DDI knowledge base. This study defined "drug-drug interaction articles" as referring only to publications that contained information about the effects of two drugs on each other's efficacies and on potential adverse effects their concomitant administration might have the patient. In particular, this definition also included articles discussing specific DDI risk factors and treatments, as well as articles disproving suspected DDIs and those reporting new adverse effects of a known interaction. For comprehensiveness, the operational definition also included drug updates and review papers (monographs), as well as drug-food and drug-herb interaction reports.

The operational definition of DDI excluded (as irrelevant to the study) publications that only discussed the impact of general drug interactions without mentioning specific drugs, patient risk factors, or DDI sequelae. These included articles about drug-drug interaction patient education, the effect of DDIs on hospital length-of-stay, the dangers of polypharmacy, drug storage concerns, and the severe financial consequences of DDIs. The definition also excluded articles about chemical interactions (e.g. pesticides, lab solutions), enzyme/protein-only studies, and computerized drug interaction prediction techniques. Other publications that did not meet the study inclusion criteria included articles about drug surveillance programs, DDI monitoring and alerting programs, and physician decision support systems. Articles about DDIs among veterinary drugs not intended for humans (e.g. drug interactions in cat shampoo, equine vaccines) were also excluded.

Constructing a Corpus

Using the above definition, project investigators first manually reviewed and classified a set of 500 MEDLINE citations in order to estimate the prevalence of DDI references in MEDLINE. To obtain a rough estimate, the investigators downloaded all MEDLINE references from April 2002 and limited these to the set of articles containing a MeSH term from the "Drugs and Chemicals" category or the words "drug" or "interaction." The study team speculated that this process would remove a significant number of negatives without discarding many positive drug-drug interaction articles. Of the remaining articles, 500 were selected at random and manually reviewed for drug interaction content. This process produced 5 DDI+ citations, suggesting a 1% prevalence

of drug-drug interaction citations. The study team decided to create a test corpus that had an enriched prevalence (10%) of DDI articles, in order to boost the positive sample for use with the SVM classifier. Therefore, project members manually created a corpus of MEDLINE references, with publication dates between 1985 and 2002, inclusive. The publication era was restricted to reduce the temporal bias resulting from yearly changes in MEDLINE indexing techniques – indexing may have produced significant discrepancies in the classification of an article published in 2004 and from a similar one published in 1970.

To generate a sufficient number of positives for the dataset, a reliable and recognized source of influential drug-drug interaction articles was necessary. The project initially began by composing a list of recently verified drug-drug interactions. However, the researchers eventually identified the institution's computerized physician order entry system (CPOE) as a good source of well-maintained and expert-reviewed drug interactions. Its drug interaction database has been manually curated by expert hospital pharmacists for more than two decades and the database of over 500 significant interactions has evolved over time to exclude false-positive warnings [104]. 150 DDIs were randomly selected from this list to serve as a collection of expert-validated drug-drug interactions.

Next, the research team transformed the list of 150 pairs of interacting medications into a set of corresponding MEDLINE references from the pre-defined study period. The investigators selected eFacts Online's *Drug Interaction Facts* database as a reputable and comprehensive source of drug information with high-quality references to support each of its drug-drug interaction fact sheets [105]. Each reference in eFacts was

listed by author, journal name, publication date, volume number and page numbers – providing sufficient information to locate the article in MEDLINE. For each of the 150 institutional CPOE-derived DDIs, the researchers copied every reference from eFacts that fell within the study timeframe and could be located in PubMed. Most eFacts references provided a direct link to the corresponding PubMed citation. For eFacts references without PubMed links and those with misdirected links, the study team attempted to locate PubMed counterparts through manual PubMed searches using author names, dates, titles and title fragments, as well as combinations of these fields. This method identified exactly 200 DDI citations.

To balance the dataset with non-DDI articles, the researchers used PubMed's search feature to download the full list of PubMed Unique Identifiers (PMIDs) for every publication from 1985 to 2002, inclusive, and then randomly selected 1800 distinct PMIDs, and labeled these as negatives. The number of these DDI negative articles chosen for each year was selected to reflect the proportion of that year's articles in the positive set. If 10% of the positive articles were from the year 2000, for example, 10% of the negatives were selected from the same publication year. Since preliminary experiments suggested a 1% prevalence of true DDI articles in MEDLINE, the research team reviewed the titles and abstracts of all 1800 randomly selected references in order to eliminate any true drug-drug interaction articles that may have been randomly included in the set. When a positive DDI article was found, it was removed from the set of negatives and replaced with a neighboring negative article from the sampling frame. This process removed 16 false negatives.

All citations were downloaded in both text and XML format using EFetch, an article retrieval tool provided by PubMed [106]. Each file was marked with its unique PubMed ID and its drug-drug interaction status (DDI+ or DDI-). The final reference dataset was composed of 1800 hand-sorted negatives and 200 expert-reviewed positives, producing a corpus of 2000 unique citations with a 10% prevalence of true positive drug-drug interaction articles.

The process of selecting DDI+ and DDI- citations for the study corpus is summarized in Figure 5.

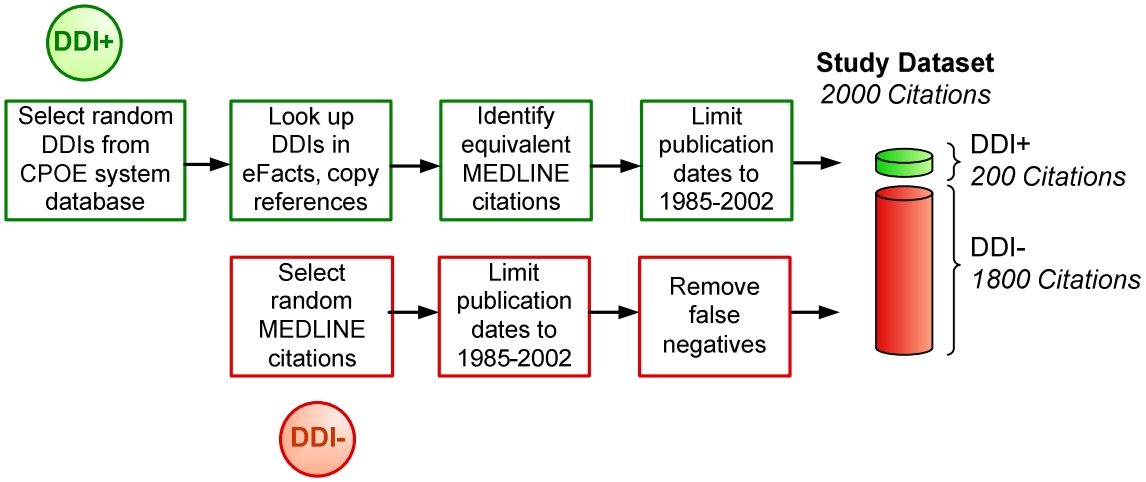

Figure 5: Corpus Construction
This image depicts the major actions performed in the construction of the study dataset. Steps for selecting DDI+ articles are shown in green; DDI- selection is in red. The final corpus was composed of 2000 citations (cylinders shown on right). One tenth of the citations were DDI+ (200 of 2000); the remaining nine-tenths were non-DDI documents (1800 of 2000).

Evaluating PubMed Using Manually Developed Queries

The National Library of Medicine (NLM) PubMed interface allows users to run complex queries against the MEDLINE database, and presents an organized set of hyperlinked results. The major search fields include the title (text words), author, abstract (text words), journal name, and publication date of a paper, as well the controlled vocabulary Medical Subject Headings (MeSH) chosen by MEDLINE indexers to characterize a paper's content [107].

The first step in the current experiment was a test of PubMed's ability to extract relevant drug-drug interaction articles from MEDLINE. Performance was measured using sensitivity (recall) and positive predictive value (precision). These measures are particularly appropriate because the value of the query results is dependent on the user's information needs. A DDI database curator looking for every publication mentioning an uncommon drug, for example, may desire high sensitivity, producing a set that may contain many irrelevant articles, but will not have missed any of the pertinent documents. On the other hand, a second user searching for information about a common drug might not want to retrieve every relevant reference (since that might be excessive), and would therefore prefer a query with high positive predictive value (PPV).

With these two information needs in mind, the researchers worked with expert librarian MEDLINE searchers from Vanderbilt's Eskind Biomedical Library to develop two baseline manual DDI queries. The first query (Q-Exp1) aimed to return a set with high sensitivity; the second query (Q-Exp2) focused on maximizing PPV. The details of these two queries are presented in Table 3.

Table 3: Two Manually-generated PubMed Queries
Table 3 lists the full text of the two expert queries used in this study.. Q-Exp1 is designed to return a large set of articles that contains every potential DDI+ article (high sensitivity). Q-Exp2 is designed to return a more limited set, in which each citation is a true DDI+ citation (high positive predictive value). These searches were developed by expert librarian MEDLINE searchers and are intended for use in PubMed's query-based search engine.

Query Name	PubMed Query
Q-Exp1 (maximize sensitivity)	("drug interactions"[TIAB] NOT Medline[SB]) OR "drug interactions"[MeSH Terms] OR drug interaction[Text Word]
Q-Exp2 (maximize PPV)	(("drug interactions"[TIAB] NOT Medline[SB]) OR "drug interactions"[MeSH Terms] OR drug interaction[Text Word]) AND ("Toxicity Tests"[MeSH] OR "Adverse Drug Reaction Reporting Systems"[MeSH] OR "Drug Hypersensitivity"[MeSH] OR "Drug Antagonism"[MeSH] OR "drugs, investigational"[MeSH] OR "Drug evaluation"[MeSH] OR "adverse effects"[sh] OR "toxicity"[sh] OR "poisoning"[sh] OR "chemically induced"[sh] OR "contraindications"[sh])

These two queries were executed through PubMed's MEDLINE interface. The set of citations returned by each query was intersected with the study dataset of 2000 references (200 DDI+, 1800 DDI-). This identified the true and false positives returned by the PubMed queries, restricted to the study DDI dataset. The project used 2x2 tables of test performance (as described in Chapter I) to calculate the sensitivity and specificity of the Q-Exp1 and Q-Exp2 search strategies.

Processing the Citation Content

In contrast to the PubMed queries, the experiments involving automated classification techniques required preprocessing of titles and abstracts. This study tested two separate methods of text preprocessing, producing two different datasets. These datasets were named CUI and TERMS.

The first dataset (CUI) was generated using a text filtering and abstracting scheme provided by the National Library of Medicine's UMLS-based MetaMap Transfer (MMTx) application (ver. AA2003). The MMTx program maps free text into UMLS concepts, and can apply this process to the titles and abstracts of MEDLINE citations. The study used a Perl script to batch process the MMTx translation of titles and abstracts from all 2000 citations in the study dataset. Within this script, each citation was processed individually using the "-a" and "-u" flags to limit acronym processing and the "-I" flag to force printing of each concept's (numeric) Concept Unique Identifier, or CUI. CUIs associated with complete UMLS mappings were retained, while candidate mappings were discarded. A binary (present/absent) vector of CUIs was created to represent the text (abstract and title) content of every citation. The set of these binary vectors for all 2000 documents constituted the "CUI dataset" that served as input for automated classification methods.

The second dataset (TERMS) was generated by extracting the title and abstract text of all 2000 corpus documents, converting text to lowercase, and replacing all punctuation with white space. The researchers also removed all stop words defined by PubMed [108]; these common words are ignored by PubMed queries and excluded from PubMed indexing. The full list of stop words appears in Table 4.

Table 4: PubMed Stop Words
The 132 common words listed below have been removed from the TERMS dataset. These stop words have little discriminatory value and have been excluded by PubMed from database searches and indexing.

PubMed Stop Words							
a	because	either	in	most	regarding	their	various
about	been	enough	into	mostly	seem	theirs	very
again	before	especially	is	must	seen	them	was
all	being	etc	it	nearly	several	then	we
almost	between	for	its	neither	should	there	were
also	both	found	itself	no	show	therefore	what
although	but	from	just	nor	showed	these	when
always	by	further	kg	obtained	shown	they	which
among	can	had	km	of	shows	this	while
an	could	has	made	often	significantly	those	with
and	did	have	mainly	on	since	through	within
another	do	having	make	our	so	thus	without
any	does	here	may	overall	some	to	would
are	done	how	mg	perhaps	such	upon	
as	due	however	might	quite	than	use	
at	during	i	ml	rather	that	used	
be	each	if	mm	really	the	using	

After removing the stop words, the remaining terms were reduced to their word stems using a publicly available Perl implementation of the Porter stemming algorithm[109, 110]. This process has been useful for preparing text for machine learning tasks [97], and is considered standard for such work.

Unlike the CUI dataset, TERMS also included the MeSH Headings and Subheadings (also known as Descriptor and Qualifier terms) associated with each MEDLINE entry. The researchers chose not to split or stem MeSH terms because they were multi-word phrases representing information content from the full text of the article. The final TERMS dataset was composed of binary present/absent vectors of these stemmed text words and MeSH terms for each of the 2000 documents in the corpus.

The process of preparing the citation text and building the CUI and TERMS datasets is summarized in Figure 5.

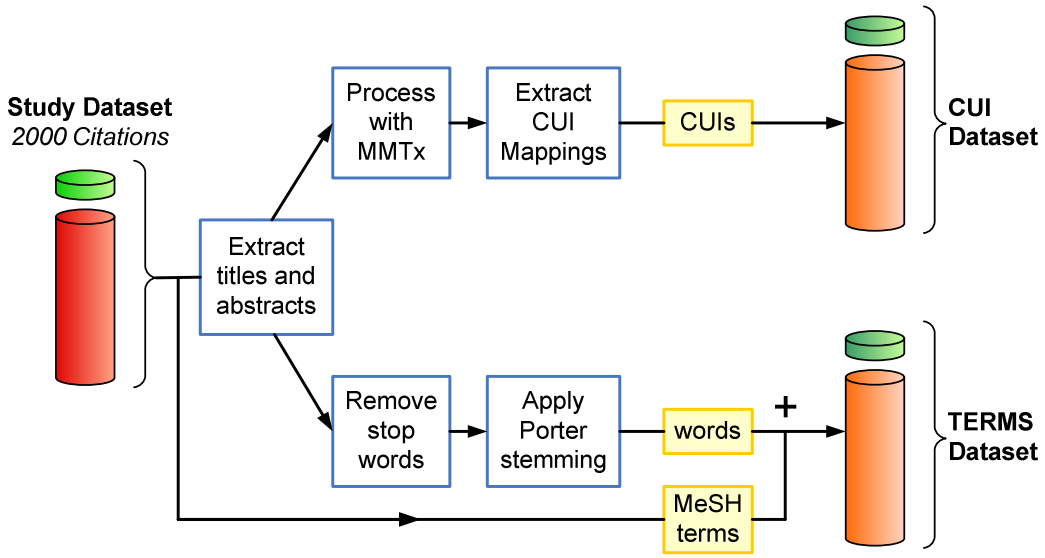

Figure 6: Generating the CUI and TERMS Datasets
The graphic above illustrates the process of converting the 2000 citations in the study dataset (cylinder on left) into the CUI and the TERMS datasets (cylinders on right). Steps in the process are outlined in blue; the final components of the CUI and TERMS datasets are outlined in yellow.

Classifying the References

The study team used the LIBSVM implementation of the SVM algorithm and conducted experiments using Matlab with a freeware SVM API available on the LIBSVM website [111]. The SVM models tested on the CUI and TERMS datasets used linear and polynomial kernels (degrees 1-4) with misclassification costs of {0.001, 0.01, 0.1, 1, 10, 100}.

The CUI and the TERMS datasets were processed independently using the same methods. One third of each dataset was put aside as a test set, and the remaining 66.7% was retained as training data, which in turn was divided into 10 mutually exclusive sets ("folds"). A 10% prevalence of positives was maintained across all the resulting sets to

ensure that results were based on the same underlying class distribution. The study used standard 10-fold cross-validation to obtain an unbiased performance estimate. By repeatedly using nine folds for training and the remaining fold as a validation set, it is possible to prevent gross overfitting of the data. The researchers plotted the ROC curve for each model and used the area under the receiver operating curve (AUC) to measure a classifier's performance. The kernel and cost parameters that produced – in a cross-validated fashion – the best AUC were used to develop a model that was tested on the previously identified and untouched test set.

The researchers also applied feature selection to the CUI and TERMS datasets to identify terms with high discriminatory power. HITON is a Markov Blanket induction algorithm recently developed by Aliferis and colleagues. Its authors have shown that over a range of tasks including text classification, HITON identifies highly discriminatory feature sets that are more compact than the sets identified by other state-of-the-art feature selection algorithms [112]. Aphinyanaphongs and colleagues have also applied HITON to reduce feature sets of text words from MEDLINE citations [103].

The HITON algorithm identifies the minimal set of variables needed to predict the target variable, *T*, in a Bayesian network. This set of nodes, known as the Markov Blanket, consists of *T*'s parents, its children, and its parents' children. In particular, this project used the HITON-MB algorithm, which seeks the full Markov Blanket, and the HITON-PC algorithm, which identifies only the set of parents and children variables in the Bayesian network. Both HITON-MB and HITON-PC were applied with and without a wrapping step that attempts to further reduce the number of features by assessing the

usefulness of subsets of variables [113]. These feature selection techniques were also performed 10 times in a cross-validated fashion.

Results

The manual review of the study dataset's 1800 randomly selected "negative" citations identified 16 with drug-drug interaction content, suggesting a DDI+ prevalence of 0.8% in MEDLINE. Although this value was slightly lower than the 1% derived from the study's initial survey of 500 citations, it was also derived from a randomly selected and larger sample (1800 vs. 500 citations). As 0.8% is the more conservative estimate of DDI+ prevalence in MEDLINE, it has been used for all future estimates of DDI+ prevalence.

Expert-generated Queries

The first "expert" query tested on the study dataset of 2000 citations was Q-Exp1. It correctly identified 150 of the 200 true DDI articles, and 1,783 of the 1,800 DDI- articles. These results are presented in classic 2x2 format in Table 5.

Table 5: 2x2 Table for Q-Exp1

	True Classification (study dataset)		
Q-Exp1 Classification	DDI+	DDI-	**Total**
DDI+	150	17	167
DDI-	50	1,783	1,833
Total	200	1,800	2,000

The sensitivity and specificity of this query were calculated using the methods described in Chapter I. This process was repeated for Q-Exp2, which identified 76 of the 200 true DDI articles and 1,798 of 1,800 DDI- citations. The data are presented below in Table 6.

Table 6: 2x2 Table for Q-Exp2

	True Classification (study dataset)		
Q-Exp2 Classification	DDI+	DDI-	**Total**
DDI+	76	2	78
DDI-	124	1798	1922
Total	200	1,800	2,000

The results of both expert-generated PubMed queries are presented in Table 7. For each query, the table lists the total number of MEDLINE articles returned, the number articles the query labeled as DDI+, the number of those which were truly positives, and the query's sensitivity and specificity.

Table 7: Performance Results for Expert Queries
The results of both expert queries are compared in this table. Best values for sensitivity and specificity are marked in bold.

PubMed Query	**Returned by query as DDI+**	**True DDI+**	**Sensitivity**	**Specificity**
Q-Exp1 *(maximize sensitivity)*	167	150	**0.7500**	0.9906
Q-Exp2 *(maximize PPV)*	78	76	0.3800	**0.9989**

Q-Exp1, the "high sensitivity" query, retrieved citations from the study corpus with a sensitivity of 0.75 and a specificity of 0.9906. Q-Exp2, the "high PPV" query identified drug-drug interaction documents with a much lower sensitivity (0.38), but a higher specificity (0.9989).

Computer Classification Models

The results of the automated classification experiments are presented for each dataset and feature selection method. For each combination, Tables 8 and 9 list the number of features in the final model (built from the entire training set) and the AUC performance on both the training and testing sets. The training set's AUC is averaged across all 10 folds of the data. Models with the highest AUC on the test set are highlighted. Table 8 displays the results of the CUI dataset.

Table 8: CUI Dataset Results
Performance of SVM models on the full CUI dataset and CUI subsets determined by four feature selection algorithms. The best performing models (highest AUC on test set) are marked in bold.

Feature Selection Method:	# Features	AUC (train)	AUC (test)
None	13187	0.9504	**0.9795**
HITON-PC	32	0.9050	0.9675
HITON-PCW	30	0.9116	**0.9705**
HITON-MB	152	0.9081	0.9616
HITON-MBW	149	0.9052	0.9474

The SVM classifier using the full set of 13187 CUIs showed the best performance, producing an AUC of 0.9795. The model identified by HITON-PCW (Par-

ents and Children with wrapping) also scored very highly, but was simpler (only 30 features) and computationally much less costly. The latter model was generated using a linear classifier with a misclassification cost of 10. The top 30 discriminatory CUIs (as selected by HITON-PCW) are listed in Table 15 in Appendix A.

While the CUI models were developed from text-to-UMLS mappings, the TERMS data included stemmed text words and MeSH terms. The results of experiments using the TERMS dataset are presented in Table 9.

Table 9: TERMS Dataset Results
Performance of SVM models on the full TERMS dataset and TERMS subsets determined by four feature selection algorithms. The best performing models (highest AUC on test set) are marked in bold.

Feature Selection Method:	# Features	AUC (train)	AUC (test)
None	22586	0.9892	0.9887
HITON-PC	13	0.9552	**0.9893**
HITON-PCW	12	0.9577	0.9860
HITON-MB	34	0.9633	**0.9900**
HITON-MBW	24	0.9668	0.9821

The full TERMS dataset included 22586 distinct word stems and MeSH terms. Of the four HITON varieties applied to reduce the number of features, HITON-PC (13 variables) and HITON-MB (34 variables) had the best classification performance. Table 16 in Appendix A lists the MeSH terms and stemmed text words selected by these two methods. The HITON-PC and HITON-MB models were both generated using a linear SVM with misclassification cost of 1.

Performance of Models vs. Queries

The AUC performance of the SVM classifiers is graphically displayed in Figure 6. The single-point performances of the two PubMed queries are annotated.

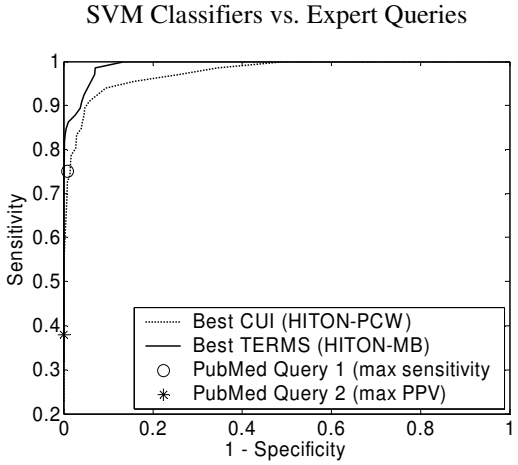

Figure 7: ROC Curves for Best SVM Models
Receiver operating characteristic (ROC) curves for the best classifiers from the CUI and TERMS dataset as well as performance points of both expert-designed PubMed queries.

Across all data points, the best TERMS model showed equal or better performance than the best model derived from the CUI dataset. Both models were able to match the performance of PubMed Query 2 (marked with an * in the graph above), which was the query designed to optimize PPV. The TERMS model outperformed PubMed Query 1 (Q-Exp1) as well.

Discussion

Principal Findings

The research presented in this chapter describes a first approach to developing an automated classifier for drug-drug interaction citations from MEDLINE. The project identified what DDI content in MEDLINE may be useful for the construction and maintenance of drug-drug interaction knowledge bases, assembled a corpus of 2000 MEDLINE citations, and produced a binary dataset of these documents for developing machine learning models.

The results of the experiments presented in this chapter indicate that SVM classifiers, when trained on a dataset of stemmed text words and MeSH terms, have the potential to perform as well as manually-generated PubMed queries in identifying articles about drug-drug interactions. Of the decision features in the smallest TERMS feature set, 9 out of 13 were MeSH headings and subheadings. These features were chosen by HITON to be highly discriminatory, which supports previous studies of the high information content of MeSH terms.

It is conjectured that the performance of the CUI-trained SVM classifiers resulted from the dataset content rather than the learning method, since the CUI dataset did not include MeSH terms (which are known to have high discriminatory power.) The TERMS-trained SVM classifiers and the manually developed PubMed queries did include MeSH content, which may have boosted their performance. Future work may explore combining CUI content with MeSH terms.

The consistently higher AUCs on the test data relative to the training data of both the TERMS and CUI datasets can be attributed to a combination of two factors. First, there is variance in the error estimates given a finite sample, and second, the features selected by the various HITON variants (PC, PCW, MB, and MBW) are not independent because of the similarities in the inductive biases of these variants.

Study Limitations

It is important to note that the two manually developed "expert" queries were created by medical librarians only, and the resulting PubMed search strategies did not undergo any testing or validation. Therefore, these strategies do not necessarily exemplify the best possible human query, nor do they represent a validated gold standard test for this dataset. A true "expert" query would be defined by a panel of clinical pharmacists (who were also experienced PubMed searchers), iteratively refined based on performance, and validated for similar performances on the study test set and MEDLINE. Using these two queries as baseline tests may have caused manual document classification to appear less capable than it actually was.

The SVM classifier, however, could also have been improved through the use of more advanced text processing and machine learning techniques. The current experiment created a simple binary (present/absent) dataset of stemmed text words and MeSH terms, although the literature suggests that text phrases, full sentences, and conceptual relationships discovered through natural language processing may provide extra discriminators for document classification [114, 115]. A dataset with feature frequencies and term weighting, as well as an SVM learner with more advanced kernels, may also

have been able to produce a more robust and generalizable SVM model with a higher AUC.

Given that this study represented only an initial test-of-concept for DDI document classification in MEDLINE, the researchers speculate that queries designed by medical librarians (who are experienced MEDLINE searchers) are sufficiently representative of high-quality manually-written queries. Although the queries used in this study may perform slightly poorer than queries designed by a panel of clinical pharmacists and other "expert" searchers, is unlikely that this disadvantage disproportionately affected the comparison of manual and automated techniques, particularly since the SVM classifier could have been improved also, by using more advanced text processing and machine learning techniques.

Another set of limitations relates to the selection of positive versus negative documents and the creation of the corpus of MEDLINE documents. The 200 DDI positives, unlike the 1800 negatives, were not selected randomly from MEDLINE, and therefore may not be representative of the full range of MEDLINE DDI+ citations. The set of positive DDI citations was identified through Vanderbilt's proprietary CPOE system's DDI database and eFacts' drug interaction reference, and thus may correspond only to the subset of all relevant DDI articles selected by Vanderbilt pharmacists and eFacts curators. Although this study did not commission an expert review of the quality of the eFacts positives, eFacts' *Drug Interaction Facts* database is considered a reputable and comprehensive source of drug information and the related references are likely to have those same qualities. For a feasibility study, these references are sufficiently representative of the high-quality drug interaction articles that are used in the creation

and maintenance of DDI knowledge bases. Extensions to this work, however, should explore a broader selection of DDI references.

There may also be content and labeling differences among articles from different years, particularly since the study corpus spans 18 years of MEDLINE (1985-2002) and the NLM makes yearly changes in its MeSH indexing strategy [58]. The result could cause disparity in the retrieval of older versus more recent articles. To minimize any indexing differences between the positive and negative sets, the proportions of DDI+ and DDI- for each year were selected to be equal. Furthermore, the NLM strives maintains the stability of its MeSH indexing system, so differences in indexing techniques are unlikely to significantly affect retrieval of older versus newer articles.

A further concern is the difference in DDI+ prevalence between the study dataset and MEDLINE. The current research used an enriched set of positives (10% DDI+ versus 0.8% in MEDLINE) to boost the performance of the SVM training algorithms. As a result, the SVM classifiers may be skewed towards data with a higher prevalence of drug interaction content, and may not maintain their performance when tested on the full content of the MEDLINE database. These possibilities are explored further in the experiments detailed in Chapter IV.

Significance of Results

The research described in this chapter suggests that SVM classifiers can be used to identify relevant drug-drug interaction information with equal or better performance than PubMed queries developed by human experts alone. The project team believes that the limitations described above did not significantly inhibit the overall study objective.

This experiment also highlights several strengths of automatic document classification. By applying SVM methods to a corpus of their previously reviewed MEDLINE citations, DDI KB curators might efficiently create and customize a DDI document locator, though such results would still require validation. These models are also easily extendable given new sample data on which to train.

Another advantage of the SVM classification model is the ability to easily tune performance based on a user's particular information retrieval needs, adjusting it towards either sensitivity (minimizing false negatives) or PPV (minimizing false positives). The concept of a precision/recall "slider" may prove useful in document retrieval tasks, allowing a user to retrieve either a large, comprehensive set or a small, precise set of articles. In settings where multiple methods are used to retrieve information, for example, users may prefer tools that deliver a reliably useful set of articles from MEDLINE to complement their other strategies. This approach is often cast as a "relevance" measure in typical information retrieval tasks, and may be worth investigating further.

Although PubMed's query-based interface is not currently configurable for automated classifiers like the one developed in this chapter, an off-site tool could download and process newly posted MEDLINE content and e-mail users if DDI+ content was detected. However, decision trees have proven useful at mapping complex classifiers such as SVMs to Boolean queries like those used by PubMed's search engine [103]. The next chapters will explore the application of decision trees and their resulting queries to drug-drug interaction article classification and the performance of these methods on the full MEDLINE database.

The CUI-based methods explored in this chapter also deserve further attention. Although CUIs have limited value for the rest of this work because they cannot be incorporated into PubMed-formatted Boolean queries, they do provide concept tagging for the title and abstract of a citation. These concept tags could provide additional information for the classification process if they could be limited to the subset of CUIs that make up the MeSH vocabulary. An algorithm developed by Bodenreider claims to find the MeSH terms most closely related to a UMLS concept [116]. By mapping CUIs back to MeSH, the concept-level knowledge identified by MMTx could be incorporated in PubMed queries. This method would take advantage of the high-level information available through text-to-CUI mapping, as well as PubMed's build-in search tools.

CHAPTER III

GENERATING QUERIES FROM SVM MODELS

Introduction

The experiments described in Chapter II determined that an SVM classifier has the potential to perform better than human-generated Boolean queries for locating drug-drug interaction citations in a labeled corpus. However, the SVM model requires significant text preprocessing and cannot currently be used with PubMed's query-based search interface. Chapter I, however, presented the approaches used by Flake and Aphinyanaphongs to develop Boolean queries styled on SVM output, a technique that may also be applicable in the DDI domain [103, 117].

The second part of this study, described in the current chapter, outlines a method of generating Boolean queries that mirror the performance of an SVM classifier. The project first constructed decision trees whose document classification approximates the output of two of the SVM classifiers developed for the TERMS dataset (HITON-MB and HITON-PC). The study team selected the tree that most closely represented its SVM parent and had the fewest features. Further steps involved decomposing the decision tree into a series of usable Boolean queries and testing the performance of these queries on the study dataset.

Methods

The second phase of the project employed the HITON-MB and HITON-PC classifiers generated from the TERMS dataset described in Chapter II. These two SVM models provided the best performance with the fewest number of features. The HITON-MB SVM classifier (34 features) predicted the DDI status of a document most accurately, as suggested by its AUC of 0.9900. The model generated from the HITON-PC variable set was the most economical classifier, with only 13 terms, and also the second best classifier (AUC = 0.9893).

Generating Decision Trees

The study team isolated the 34 variables identified by the HITON-MB feature selection algorithm, as well as the vector of predictions (output) produced by the HITON-MB SVM classifier. These data were used as predictors (34 variables) and response values (1 vector) to construct a preliminary unpruned decision tree using the `treefit` decision tree regression algorithm from Matlab's Statistical Toolbox [118]. Given only the training data, the researchers used a 10-fold cross-validation method to evaluate alternative pruned trees and estimate the optimal pruning level (`treetest`). The best pruning level was considered to be the one that produced the smallest tree whose cost was within one standard error of the minimum cost. The preliminary unpruned tree was then pruned to the optimal pruning level, using Matlab's `treeprune`, producing a sparse version of the decision tree. This process was repeated with the HITON-PC SVM classifier (13 features, 1 prediction vector) to generate the decision tree equivalents of

both SVM models. The structures of both decision trees are depicted in Figure 9 and Figure 11 of Appendix B.

The study measured the similarity of each decision tree model to its parent SVM classifier by comparing the decision tree classifications to the SVM predictions and calculating the area under the receiver operating curve (AUC). AUC values closer to 1 indicated greater similarity between the two classifying methods. Each decision tree was used to classify both the training and test data, and appropriate AUCs were computed. The simplest decision tree that most closely modeled the classification capabilities of its SVM parent was chosen for further experiments.

Evaluating Performance Thresholds

The leaf nodes of the selected decision tree have values that are continuous. In a further step towards generating Boolean queries, this single continuous-valued tree was converted into many binary trees. To produce a binary decision tree, every leaf node with a score greater than or equal to the threshold score was assigned a value of TRUE (1), signifying a prediction of DDI+. Leaf nodes with values below the threshold score were labeled FALSE (0). Each unique leaf node value was used, in turn, as a threshold score in this conversion process.

Each binary decision tree was used to classify the test set, and its performance (sensitivity and specificity) was recorded. The research team chose three binary trees with different sensitivity-specificity combinations to convert to Boolean queries.

Designing Boolean Queries

Each of the three binary decision trees was converted to a Boolean query by stringing together the text elements for every DDI+ (TRUE) path. The terms along a single TRUE root-to-leaf path were joined with AND or NOT statements, as appropriate. The resulting paths were concatenated using OR statements to produce a Boolean matching pattern for DDI+ citations. This process was repeated for each of the three binary decision trees, producing three distinct Boolean queries.

Queries were adapted for PubMed by adding a "[MeSH]" tag for MeSH terms and the "[sh]" tag for MeSH subheadings. AND statements directly preceding NOTs were dropped and any stemmed text words included in the query had an asterisk (*) appended to include word variants.

Evaluating Query Performance

In a final step, the researchers tested the performance of all five queries (2 expert-written, 3 computer-generated) on the study dataset. Each query was executed on the MEDLINE database, using the PubMed interface to limit the query to the same timeframe as the study dataset (1985-2002.) Articles returned by the query were considered DDI+ for that query method; all others were labeled DDI-. The set of citations returned by each query was intersected with the 2000 hand-labeled true positive and true negative citations to determine the query's performance on the study dataset.

Results

Performance of Decision Trees vs. SVMs

The first of the two decision trees, Tree PC, was generated from the HITON-PC classifier and involved only 13 features. Tree MB was derived from the HITON-MB classifier (34 features). The details of Trees PC and MB are listed in Table 10, along with an AUC measure of how closely each tree models the behavior of its SVM parent. Decision tree AUCs are listed for both the training and the test data. Illustrations of Tree MB and Tree PC are provided in Figure 9 and Figure 11 of Appendix B.

Table 10: Results of Decision Tree Construction
Table 10 lists the classification performances of Tree PC and Tree MB, the feature selection method of its SVM parent, and the size of its resulting feature set. Columns on the right list how well each SVM parent classified the test DDI data, and how closely the derived decision tree (DT) modeled its SVM parent on the training and test sets, respectively. The best values in each category are marked with bold text.

			Performance of SVM Parent	**DT modeling of SVM classifier**	
Decision Tree	**Feature Selection Method**	**# Features**	**AUC (test set)**	**AUC (training set)**	**AUC (test set)**
Tree PC	HITON-PC	**13**	0.9893	**0.9879**	**0.9768**
Tree MB	HITON-MB	34	**0.9900**	0.9864	0.9766

Both Tree PC and Tree MB closely reproduced the output of their parent SVM classifiers, with AUCs of 0.9768 and 0.9766 on the testing data, respectively. With such similar performance, however, the simplicity of the HITON-PC feature set made Tree PC the more appealing choice.

Selected Performance Thresholds

Tree PC produced 48 unique leaf node values (threshold scores), resulting in 48 binary trees. Each binary decision tree performed differently when classifying the study dataset. The research team selected three representative trees, Binary DT-3, Binary DT-4, and Binary DT-5, generated from the threshold values -0.9875, 0.4360, and 0.8560, respectively. Figures 12 through 14 of Appendix B show these three binary trees projected in color onto the Tree PC structure.

The sensitivity and 1-specificity of all 48 trees are plotted in Figure 7, with markers and threshold scores for the three selected trees.

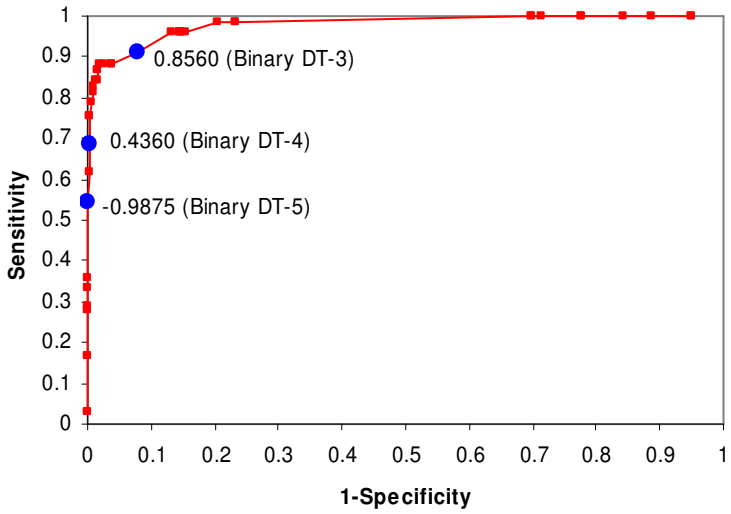

Figure 8: Performance of 3 Binary Decision Trees
The performances of all possible binary decision trees derived from Tree PC are plotted as points in the graph above. The three binary trees selected for query conversion are marked with larger points in blue and are labeled with their threshold scores and names.

A table of all of Tree PC's 48 threshold values and their related sensitivities and specificities, which includes all unmarked points in Figure 7, can be found in Table 17 of Appendix A.

Table 11, shown below, lists the three selected binary decision trees, the threshold values that generated them, and their performances on the training dataset (1334 examples). These three decision trees represent different sensitivity-specificity combinations, as evidenced by the graph above: balanced sensitivity/specificity, high sensitivity with low specificity, and very high sensitivity with very low specificity.

Table 11: Selected Binary Decision Trees and Performance Scores

Tree	Threshold Value	Sensitivity	Specificity
Binary DT-3	-0.9875	0.9104	0.9208
Binary DT-4	0.4360	0.6866	0.9983
Binary DT-5	0.8560	0.5448	0.9992

Binary DT-3 was generated by polarizing scores on each side of the -0.9875 threshold. This tree showed a balanced performance on the study test set, with a sensitivity of 0.9104 and a specificity of 0.9208. Binary DT-4 and Binary DT-5 were generated from the higher threshold scores of 0.4360 and 0.8560, respectively. These two trees classified the citations from the study dataset with higher specificity, but lower sensitivity.The pruned versions of these three binary decision trees are shown in Figure 15 (a) through (c) of Appendix B.

Three Computer-generated Queries

Each of the three decision trees was successfully decomposed into a PubMed-formatted Boolean query.

Table 12: All Computer-generated Queries
All three computer-generated queries developed as part of this research are listed in Table 12. These search strategies are intended for use in PubMed's query-based search engine.

Query Name	Query Detail
Q-Comp3 (computer-generated query from Binary DT-3)	"drug interactions"[MeSH] OR ("humans"[MeSH] AND "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND "adverse effects"[sh] AND receiv* NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND "adverse effects"[sh] AND interact* NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH] NOT receiv*) OR ("humans"[MeSH] AND "adverse effects"[sh] NOT interact* NOT cell* NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH] NOT receiv*) OR ("humans"[MeSH] AND "pharmacology"[MeSH] NOT cell* NOT "adverse effects"[sh] NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND receiv* NOT cell* NOT "pharmacology"[MeSH] NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH] NOT "adverse effects"[sh]) OR ("pharmacology"[MeSH] AND interact* NOT cell* NOT "humans"[MeSH] NOT "drug interactions"[MeSH]) OR (interact* AND "drug synergism"[MeSH] NOT "pharmacology"[MeSH] NOT cell* NOT "humans"[MeSH] NOT "drug interactions"[MeSH])
Q-Comp4 (computer-generated query from Binary DT-4)	("drug interactions"[MeSH] AND "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND interact* NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND receiv* NOT interact* NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND "humans"[MeSH] NOT receiv* NOT interact* NOT "pharmacokinetics"[MeSH])
Q-Comp5 (computer-generated query from Binary DT-5)	("drug interactions"[MeSH] AND "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND interact* AND "pharmacology"[MeSH] NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND interact* AND "adverse effects"[sh] NOT "pharmacology"[MeSH] NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND receiv* NOT interact* NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND "humans"[MeSH] AND "pharmacology"[MeSH] NOT receiv* NOT interact* NOT "pharmacokinetics"[MeSH])

Binary DT-3 produced Q-Comp3, the first of the computer-generated queries. The TRUE branches of decision tree Binary DT-4 were concatenated to produce Q-Comp4, a second computer-generated query. Binary DT-5 produced Q-Comp5. All three queries are detailed in Table 12.

All three queries could be reduced in length and complexity using first order logic, but they have been retained as disjunctions of conjunctions to illustrate how they were derived from decision trees. The PubMed query engine ably handles either format.

Performance of Expert and Computer-generated Queries on Study Dataset

All 5 study queries were tested on the study dataset of 2000 citations (200 DDI+, 1800 DDI-) and the results are presented in Table 13. Citations returned by a query were considered positive (DDI+); all others were labeled as negatives (DDI-). Table 13 lists the size of each query's return set, how many of those citations were true versus false positives, and the calculated sensitivity and specificity of each query. Q-Exp1 and Q-Exp2 refer to the two expert-written queries described in Chapter II. Q-Comp3, Q-Comp4, and Q-Comp5 are the names of the three computer-generated queries listed above.

Table 13: Results of 5 Queries on the Study Dataset of 2000 Citations
This table shows how accurately the study's two expert queries (Q-ExpN) and three computer queries (Q-CompN) classified the 200 DDI+ and 1800 DDI- citations. The better classifiers have true positive values close to 200 and false positives values close to 0. Sensitivities and specificities closer to 1 indicate better performance.

	Size of the query's return set (TP + FP)	**True Positives** (out of 200)	**False Positives**	**Sensitivity**	**Specificity**
Q-Exp1	167	**150**	17	**0.750**	0.991
Q-Exp2	78	76	**2**	0.380	**0.999**
Q-Comp3	345	**185**	160	**0.925**	0.911
Q-Comp4	158	149	8	0.745	0.996
Q-Comp5	91	88	**3**	0.440	**0.998**

Expert query 1 (Q-Exp1) detected 150 of the 200 true positives in the study dataset, and had a specificity of 0.991. Q-Exp1 is a very broad, comprehensive query, so it is of note that a computerized query (Q-Comp4) was able to achieve a higher sensitivity. On the other hand, Q-Exp2 showed the poorest sensitivity (0.380) of any query, but paired this with the highest specificity (0.999).

Q-Comp3 detected the most DDI+ citations of any query, with a high sensitivity of 0.925. It also returned the highest number of false positives, leading to a lower specificity of 0.911. Computer query 4 (Q-Comp4) correctly classified 149 of 200 true positives, only one citation fewer than Q-Exp1. The query's specificity, however, was significantly higher (0.996), since it detected fewer than half the false positives. The query Q-Comp5 produced the second smallest return set, the second lowest sensitivity (0.440) and the second highest specificity (0.998), with only a single false positive more than Q-Exp2.

The computer-generated query Q-Comp4 and the expert query Q-Exp1 showed comparable abilities for detecting DDI+ articles in the study dataset. Q-Exp1 identified an additional positive but statistically insignificant citation (150 DDI+ vs. 149 for Q-Comp4, $x^2 = 0.00658$, $p = 0.01$). However, Q-Comp4 identified 9 fewer false positives than Q-Exp1, a difference which did prove to be statistically significant at $p = 0.01$ ($x^2 = 9.07157$). Q-Exp2 and Q-Comp5 present a similar case.

Discussion

Principal Findings

The results of these experiments suggest that one can derive Boolean queries that accurately model the performance of SVM classifiers, and that these queries can perform at least as well as manually developed queries when identifying drug-drug interaction articles in a study dataset. Each manually-generated query's performance was well-matched and possibly exceeded by that of a computer-generated query, suggesting that it is possible to achieve acceptably functioning queries using computer-assisted methods rather than human effort alone.

The slight differences in the sensitivities and specificities of the computer-generated queries when compared to decision tree's classification performances are likely due to differences in the makeup of the dataset used; the former method was tested only on the training data, while the latter evaluation took place on the full set of 2000 citations. Within the training set, both procedures identified the same set of articles.

Study Limitations

The switch from SVM classifier to Boolean query represents a probable decrease in performance, since the decision tree lacks the flexibility of an SVM model. In addition, the SVM classifier is easily tuned for high sensitivity or high positive predictive value, whereas the query represents static performance. The experiment described in this chapter, however, represents a necessary step if SVM-based classification is to be integrated into existing search engines. The process of converting machine learning classifiers to simple queries aims to increase the practicality and usability of classifiers, and leverage the power of these computer-based methods for locating relevant content in MEDLINE.

A further concern is the difference in DDI+ prevalence between the study dataset and MEDLINE. This study used an enriched set of positives (10% DDI+ versus 0.8% in MEDLINE) to boost the performance of the SVM training algorithms. As a result, the SVM classifiers may be skewed towards data with a higher prevalence of drug interaction content, and may not maintain their performance when tested on the full content of the MEDLINE database. These possibilities are explored further in the experiments detailed in Chapter IV.

Significance of Results

This series of experiments supports previous research on the successful translation of SVM models to Boolean queries, and applies this method to the specific area of drug-drug interaction information in MEDLINE. This research emphasizes that when choosing

a query threshold for a decision tree, one must allow for slight changes in performance when the resulting model is applied to unseen examples from the same population.

The three queries developed in these experiments have performed at least as well as the two expert-written queries described in Chapter II. One might expect that computer-generated queries have the potential to outperform manually generated queries, if correct threshold values can be identified and appropriate queries generated. It is likely that decision trees generated from larger feature sets – unlike Tree PC's small set of 13 features – will offer more unique leaf values (threshold scores), allowing them to generate more Boolean queries of different performance potential.

CHAPTER IV

COMPARING MANUALLY-GENERATED QUERIES WITH COMPUTER-GENERATED QUERIES IN MEDLINE

Introduction

The experiments of the previous chapter have determined that computer-generated Boolean queries have the potential to perform better than manually created queries at identifying drug-drug interaction citations. This study used a corpus of "positive" and "negative" DDI citations to generate datasets composed of MeSH terms, CUI-tagged title and abstract text, and stemmed text words. The research team modeled the patterns in the data using an SVM classifier, mapped this classifier to a decision tree, and decomposed the tree into three PubMed-formatted Boolean queries. These three computer-generated queries were compared to two queries developed by Vanderbilt University's library staff.

In the experiments of Chapter III, the computer-generated queries displayed excellent performance given a 10% prevalence of DDI+ citations in the study dataset. Unfortunately, this performance may not carry over to MEDLINE, where only an estimated 0.8% of content is related to drug-drug interactions.

The final experiments of this study evaluated the performance of all five queries on a full year of MEDLINE citations to determine their actual classification capabilities. The details of all five queries are assembled for review in Table 18 of Appendix A.

Methods

All five expert- and computer-generated queries were executed on the MEDLINE database, using PubMed's query interface. The return set was limited to articles published in 2003, a year purposely chosen to be outside the timeframe of the study dataset (which spanned 1985 to 2002). 400 articles were randomly selected from each query's MEDLINE 2003 return set and manually classified by one reviewer according to their drug-drug interaction applicability, per the definitions of Chapter II.

The researchers calculated the positive predictive value of each query and the 95% confidence interval associated with each query's return set. Each query's sensitivity, specificity, NPV, and PPV were computed and these numbers were used to evaluate each query's performance.

Results

At the time of this study, MEDLINE had indexed 579,884 citations from 2003. This value and Chapter II's estimate of the prevalence of DDI+ articles in MEDLINE (0.8%) were used to determine approximate sensitivities, specificities, and negative predictive values for the five queries.

Table 14 lists the number of MEDLINE articles published in 2003 that each query returned as positive, the number of true and false positives found in the survey of 400 randomly selected query citations, and the resulting positive predictive value (PPV). Table 14 also presents the estimated sensitivity, specificity, and negative predictive value (NPV), which have been calculated by assuming a 0.8% prevalence of DDI+ articles in

MEDLINE. This approximate value was derived from the survey of citations described in Chapter II.

Table 14: Performance of All Queries on MEDLINE 2003
This table shows how effectively the study's two expert queries (Q-ExpN) and three computer queries (Q-CompN) identified DDI+ articles among all MEDLINE articles published in 2003. True positive values close to 400 and false positive values close to 0 are desirable. Better classification is indicated by sensitivity, specificity, PPV and NPV values closer to 1. The highest sensitivities and positive predictive values for each category (expert, computer) are marked in bold.

	MEDLINE articles returned	**True Positives** in 400	**False Positives** in 400	**PPV**	**NPV** (estimated)	**Sensitivity** (estimated)	**Specificity** (estimated)
Q-Exp1	4,411	224	176	**0.56** (± 0.05)	>0.99	**0.53**	>0.99
Q-Exp2	1,581	218	182	0.54 (± 0.04)	>0.99	0.19	>0.99
Q-Comp3	61,398	46	354	0.12 (± 0.05)	1.0*	**0.90**	1.0*
Q-Comp4	3,768	211	189	0.53 (± 0.05)	>0.99	0.43	>0.99
Q-Comp5	1,370	255	145	**0.64** (± 0.04)	>0.99	0.19	>0.99

Given the return size of Q-Comp3 (61,398 citations) and its PPV (0.12), calculations suggest this computer-generated query retrieved more true DDI+ citations than actually exist in 2003 MEDLINE, as estimated by the 0.8% prevalence measure. Performance estimates for which these numbers presented mathematical complications have been marked with an asterisk (*). Although it is unlikely Q-Comp3's true specificity and NPV are 1.0, suggesting perfect exclusion of negatives, the query's performance in these categories appears likely to approach these values, outstripping the other queries. The query's estimated return of true DDI+ citations, rather than the 0.8% prevalence of DDI+, was used to set a conservative lower bound on sensitivity (0.90).

The PPVs for the two expert queries Q-Exp1 and Q-Exp2 were similar to each other, with values of 0.56 and 0.54, respectively. The queries' sensitivities reflected the size of their return sets, however, with an estimated sensitivity of 0.53 for Q-Exp1 and 0.19 for Q-Exp2.

Q-Comp3, as discussed above, posted a high sensitivity, but an extremely low positive predictive value (0.12), suggesting that only 1/10 of this query's return set would offer useful DDI information. A middle-of-the-road query in terms of performance, Q-Comp4 fell behind Q-Exp2 with a slightly lower PPV (0.53 vs. 0.56) and sensitivity (0.43 vs. 0.53). On the other hand, Q-Comp5 matched Q-Exp2 for sensitivity (0.19), but posted the highest PPV of any query (0.64), outstripping Q-Exp2's PPV of 0.54.

Discussion

Principal Findings

As discussed in Chapter II, the value of a query's results is dependent on the user's information needs. Therefore, "good performance" can be a measure of high sensitivity (recall), high PPV (precision), or a combination of the two values. This experiment showed that when performance is measured by sensitivity, the expert-generated queries may be the more efficient choice. However when performance is measured by PPV, computer-generated queries can rival human queries: on the MEDLINE 2003 dataset, Q-Comp5 presented the highest PPV of any query. There is some overlap, however, among the 95% confidence intervals of all four of these queries,

so it is not possible to distinguish whether the computer-generated or traditional queries performed significantly better overall.

The reported values of negative predictive value, sensitivity, and specificity are only estimates. The uncertainty in these values combines the possible variation in PPV as well as the unknown error in the current study's estimate of DDI+ prevalence in MEDLINE. The computer-generated query Q-Comp3, however posted a much higher sensitivity (0.90, estimated) than any of the other four queries. Although this high sensitivity was also paired with the lowest PPV, Q-Comp3 might be the best choice for a user looking to capture all possible DDI+ articles, with no concern for false positives.

It is interesting to note that many citations containing valuable DDI+ content do not include the text phrase "drug interaction" and are not tagged with the MeSH term "Drug Interactions." Q-Exp1 was a very broadly stated query based on these assumptions, and yet its sensitivity was only 0.53 on the MEDLINE 2003 set, suggesting it retrieved barely over half the relevant DDI+ articles. While MeSH terms can clearly be used to locate DDI information (9 of the 13 highly discriminatory features identified by HITON-PC were MeSH terms), the necessary MeSH terms are not always clearly related to the target content. Of the MeSH terms identified by HITON-PC, for example, "Hematoma, Subdural" is not intuitively connected to drug interactions. This study attempted to find alternate methods of identifying significant articles, but improvements in title and abstract formulation and MeSH indexing also might assist in the identification of relevant information.

Study Limitations

The applicability of these study results is limited, as the study did not manually review the full return set of each query and the process was completed with only a single reviewer. A survey of 400 randomly selected articles was sufficient to determine with 95% confidence the prevalence of DDI+ articles within 5%. The resulting overlap in confidence intervals, however, leaves it impossible to definitively state which query had the best performance, as determined by PPV and sensitivity. It is clear that for queries QExp1, Q-Exp2, Q-Comp4, and Q-Comp5, these differences are small, since all four queries registered PPVs in the 0.55-0.65 range. This suggests that computer-generated queries and expert queries can produce return sets with similar true-positive rates. Indeed, it is possible that a user's search experience may not be affected by small variances in PPV.

Significance of Results

Although MEDLINE undergoes yearly indexing changes, it is unlikely that citations retrieved from 2003 were significantly different those in the training set (1985-2002). Nevertheless, all five queries posted poorer overall performances on MEDLINE than on the study dataset, which reinforces the assumption that the study dataset, with its enriched set of positives, was not entirely representative of raw MEDLINE content. The experiments described in this chapter, therefore, comprise a necessary validation step, since SVM models are known to be sensitive to prevalence and the SVM-based classifiers developed in Chapters II and III used a dataset with a higher percentage of DDI+ citations than MEDLINE. Testing these queries on PubMed ensures that this

method of developing an SVM classifier and decomposing it into a series of Boolean queries is, despite a decrease in performance, still compatible with MEDLINE.

The results of this experiment confirmed that in the drug-drug interaction domain, computer-generated queries can rival manually created queries for precision (PPV), and may outperform these queries when performance is measured by recall (sensitivity).

CHAPTER V

SYNOPSIS AND CONCLUSIONS

Summary

This study developed and evaluated a novel approach to locating drug-drug interaction MEDLINE content, using computer-generated queries in place of traditional expert PubMed queries. This report described the method in three stages: Chapter II presented the construction of an SVM classifier using a dataset of 2000 MEDLINE citations; Chapter III discussed decomposing this classifier into a series of Boolean queries; and Chapter IV tested these queries on one full year of MEDLINE citations.

The SVM classifier described in Chapter II proved to be an effective tool for identifying MEDLINE citations with drug-drug interaction content, particularly when the model was trained on a dataset of stemmed text words and MeSH terms. When matched for specificity, the SVM model consistently surpassed the two expert queries in sensitivity, returning a greater number of DDI+ articles from the study dataset.

Chapter III described the process of converting an SVM model to a decision tree, and presented two decision tree classifiers that closely resembled their SVM antecedents. These decision trees were easily broken down into Boolean queries. When compared to the two queries produced by biomedical librarians, the computer-generated queries demonstrated equal or better sensitivities and specificities on the study dataset of 2000 articles. Although the SVM model was a more accurate classifier, these queries did not

require document pre-processing and could be executed quickly using PubMed's search interface.

In Chapter IV, the current study explored the application of these search strategies to a full year of MEDLINE citations. Both computer-generated and expert queries displayed a significant drop in performance when tested on the MEDLINE database versus the study dataset, which was not surprising given MEDLINE's low prevalence of DDI+ citations. Nevertheless, the results of this experiment indicated that computer-generated queries can rival human queries when used to identify drug-drug interaction content in MEDLINE. In addition, these computer-based queries may prove to be better at accommodating users' varied information retrieval needs.

A noteworthy strength of the approach taken in this work is that a single SVM model can be used to generate multiple Boolean queries whose performance is predictable. Manually-generated queries, by contrast, must be constructed individually and tested on a sample dataset in order to evaluate their performance. In Chapter II, the flexibility of the SVM model was described as a sensitivity-specificity "slider," which would allow users to specify a particular classification performance level by choosing a point along the ROC curve. Converting the SVM classifier to a decision tree resulted in the discretization of the values along the ROC curve, producing a series of "threshold scores." Although these scores represent a more limited selection of potential sensitivity-specificity combinations than the "slider" model, they can be used in conjunction with a decision tree to produce many Boolean queries of varying sensitivity and positive predictive value. This technique offers the user a broader range of performance choices than a single expert-designed query.

Study Limitations

It is crucial to consider the various limitations of the current work, especially in its design and selection of queries, and its methods of dataset construction. These factors affect the study's external validity.

"Expert" Queries

This study compared only five queries for locating DDI+ articles. Testing a greater number of both human and automated queries would allow a more thorough analysis of each approach and the quality of information it provides. Similarly, the two "expert" queries came from a single group – Vanderbilt's biomedical librarians – and do not represent the full potential of human query-writing capabilities. Moreover, the librarians did not refine the queries over time on varying MEDLINE content.

On the other hand, the machine learning models used to generate the computer-based classifiers were comparably straightforward; the study did not investigate the discriminatory power of more advanced SVM algorithms, nor were the corpus-derived datasets refined using advanced text processing methods. Neither the hand-written nor the decision tree-derived queries were tuned to optimize performance, despite studies that have shown that query expansion may, in some instances, improve the performance of MEDLINE searches [119, 120].

It was not, however, the intent of the study to conduct a rigorous analysis of human versus computer-aided query design. The research comprised an initial effort designed to explore the feasibility of a new method of query design in the drug-drug interaction domain. Results indicate that machine learning techniques can be used to

produce queries that compare to those developed by experienced MEDLINE searchers, and that DDI information retrieval methods have potential for further improvement.

Study Dataset

A clear avenue of improvement lies in developing a larger, more representative datasets with additional features. Although the study dataset of 2000 citations was sufficiently diverse to serve as a training corpus for a feasibility study, it was not adequately representative of MEDLINE to be considered a solid gold standard. The research team chose an enriched set of positives (10% DDI+ versus 1% in MEDLINE) to boost the performance of the training algorithms. As a result, the study's SVM models may have been skewed towards data with a higher prevalence of drug interaction content, and the resulting computer-generated queries may have had built-in assumptions about DDI+ prevalence.

The study used only stemmed text words and MeSH terms from MEDLINE titles and abstracts as input features for the TERMS-based SVM classifier. These features may have been insufficient for optimal text classification. A 2002 study by Ding and colleagues suggests that word fragments alone are too small of a segment of text for optimal classification; adding phrases might yield better precision [114]. Word frequency counts and weighting schemes have been used successfully in other studies that applied SVM classifiers, and similar techniques might improve upon the results presented here[121].

Furthermore, this research assumed that all DDI+ articles *can* be located by the information in their titles, abstracts, and MeSH terms. It is possible, however, that the full

range of content in an article is not evident from its MEDLINE citation alone. If this is the case, analysis of the full text of articles would be necessary to correctly identify many articles with valuable drug-drug interaction content. Although this is a highly relevant question in terms of complete DDI+ retrieval from MEDLINE, it would not have affected the current study results. Since the gold standard was established by a manual review of titles and abstracts rather than full-text documents, any DDI+ documents whose drug-drug interaction content was not apparent in the title or abstract would have been missed by the study's manual reviewers as well as the machine learning classifiers. Nevertheless, it would be preferable to use the full text of each article, but for most journals this content is not yet freely available.

MEDLINE

A more noteworthy assumption is that PubMed queries have the potential to identify every article in MEDLINE. A study by Balas and colleagues explored simple PubMed searches as a means of identifying information about health care quality improvement. Their results suggest that PubMed queries are flawed as a method of information retrieval, and that searches using MeSH terms and text words result in only moderate recall and precision [122]. However, work by Backus and colleagues has indicated that the "Drugs and Chemicals" category is one of the largest, but least often searched MeSH term sets, suggesting there is room for improvement in drug-related queries that use headings from this branch of the MeSH vocabulary [123]. Regardless, the PubMed search tool is a quick and publicly available means of searching the MEDLINE database, which is why this study has explored its use.

A particular limitation of using MEDLINE as a source of drug-drug interaction content stems from the delay between the publication of a relevant article and its indexing in the MEDLINE database. Most DDI knowledge bases are only updated once every 2-4 months, however, which leaves plenty of time for high-quality articles to enter the MEDLINE database [124]. Nevertheless, DDI KB developers in need of timely drug interaction information might find MEDLINE insufficient for their information retrieval needs. In order to obtain both high-quality and up-to-the-minute DDI information, it may be necessary to consult a variety of content sources.

Study Implications and Future Work

Although neither manual nor computer-aided queries were optimized, the current study represented a successful test-of-concept for quasi-automated DDI document classification in MEDLINE. The computer-generated queries designed in this study are comparable to traditional, expert-designed queries and may be easier and less expensive to maintain.

In addition, the approach suggests that it might be straightforward for drug database curators to develop computer-generated queries given the selection of drug-drug interaction articles they have already identified. By collecting the related citations from MEDLINE, they are guaranteed a set of useful drug interaction positives for their training and test sets.

More extensive and applied experiments will be required, however, before anyone can generate a practical classifier for identifying drug-drug interaction articles in

MEDLINE. Future studies should involve a larger training set that is more representative of MEDLINE content as well as additional examples of expert and computer-generated queries. True "expert" queries must be used to accurately evaluate the performance of traditional search strategies.

Likewise, different SVM techniques and modified datasets with word frequencies, weights, and semantic content could be used to produce a computer-generated classifier with higher sensitivity and PPV. In particular, probabilistic SVMs, which output probabilities of class assignment rather than target values, might be used to adjust for the change in DDI+ prevalence between the study dataset and MEDLINE [125, 126]. An immediate extension to the current work would train such a probabilistic SVM on the TERMS dataset and use it to generate a new set of queries that could be compared to the expert and computer-generated queries developed in the current study. A further experiment should investigate sources of false positives and false negatives, and test how the classifiers may be tuned to avoid misclassification while preserving generalizability.

Furthermore, a more helpful information retrieval system might rank citations according to their probability of containing DDI+ information, rather than offering the stark yes-or-no judgments of binary classifiers. Indeed, this approach might be more compatible with users' differing information needs, allowing them to peruse as few or as many documents as they desire and thereby creating a personal "viewing set" with either high sensitivity or high PPV. Such practical applications of the techniques presented in this report could be refined from ongoing feedback from real-world use by experts engaged in DDI identification.

Subsequent work should also take into account that MEDLINE is only one potential source of drug-drug interaction information. It may be the case that the type of information in other DDI sources (e.g. drug company publications, FDA warnings, pharmaceutical bulletins) is more useful in constructing DDI knowledge bases. Future work should investigate alternative drug knowledge sources and explore mining them for DDI content.

Conclusion

The work presented here represents a feasibility test for quasi-automated drug-drug interaction reference identification. The study results, obtained through a series of experiments, show that support vector machine classification can be used to produce Boolean queries that target DDI citations, and that these queries perform well on the MEDLINE database when compared to human experts. By enhancing MEDLINE's value as a source of drug-drug interaction information, the project team hopes to improve the accessibility of high-quality DDI content and encourage a more fact-based approach to the development of drug-drug interaction knowledge bases and alerting systems.

APPENDIX A

SUPPLEMENTARY TABLES

Table 15: Top 30 Discriminatory CUIs Selected by HITON-PCW

CUI Name	CUI
Administration	C0001554
Arrhythmia	C0003811
Carbamazepine	C0006949
Containing	C0332256
Cyclosporine	C0010592
DEBILITATION	C0742985
Dipyrone	C0012586
Direct type of relationship	C0439851
Drug Interactions	C0687133
Drug Kinetics	C0007634
Ergotism, NOS	C0595996
Erythromycin	C0014806
Flecainide	C0016229
Fluoxetine	C0016365
Flurbiprofen	C0016377
Mazes	C0870866
Nicotinic Acids	C0028049
physiological aspects	C0031843
Prolonged QT interval	C0151878
Protein measurement	C0202202
Reception	C0544683
Rhabdomyolysis	C0035410
Risperidone	C0073393
Stimulation - action	C0441691
Techniques	C0449851
Terfenadine	C0085173
Theophylline	C0039771
Topical Ointment	C0991554
Tramadol	C0040610
Warfarin	C0043031

Table 16: Features from the TERMS Dataset Selected by HITON-PC and HITON-MB
Features in bold were selected by both HITON-PC and HITON-MB. All other stemmed text words and MeSH terms were selected by HITON-MB only. The last row lists the total number of features selected by each method.

	Feature Selection Method	
Stemmed text words and MeSH terms	**HITON PC**	**HITON MB**
bind		x
cell	x	x
dietari		x
dimer		x
dose	x	x
drug		x
given		x
group		x
human		x
induc		x
interact	x	x
MeSH: administration & dosage	x	x
MeSH: adverse effects	x	x
MeSH: Animals		x
MeSH: antagonists & inhibitors		x
MeSH: Drug Interactions	x	x
MeSH: Drug Synergism	x	x
MeSH: drug therapy		x
MeSH: Drug Therapy, Combination		x
MeSH: etiology		x
MeSH: Hematoma, Subdural	x	x
MeSH: Human	x	x
MeSH: Infant, Newborn		x
MeSH: metabolism		x
MeSH: Pharmacokinetics	x	x
MeSH: Pharmacology	x	x
MeSH: Receptors, Retinoic Acid		x
MeSH: Renal Dialysis		x
MeSH: Theophylline	x	x
MeSH: therapeutic use		x
MeSH: therapy		x
MeSH: Tritium		x
receiv	x	x
undertook		x
Number of features	**13**	**34**

Table 17: All Threshold Values of Tree PC and Related Sensitivities and Specificities
Threshold values in bold were used to generated the binary decision trees DT-3, DT-4, and DT-5. Grayed-out threshold values produce decision trees that are dominated by other selections on the list.

Threshold Value	Sensitivity	Specificity
-2.2271	1	< 0.3033
-1.8222	1	< 0.3033
-1.8199	1	< 0.3033
-1.8195	1	< 0.3033
-1.8123	1	< 0.3033
-1.7983	1	< 0.3033
-1.4112	1	< 0.3033
-1.4111	1	< 0.3033
-1.4110	1	0.3033
-1.4100	0.9851	0.7658
-1.4051	0.9851	0.7658
-1.3967	0.9851	0.7942
-1.3801	0.9627	0.8458
-1.3595	0.9627	0.8525
-1.3202	0.9627	0.8525
-1.1605	0.9627	0.8542
-1.0006	0.9627	0.8558
-0.9996	0.9627	0.8692
-0.9875	**0.9104**	**0.9208**
-0.9846	0.9104	0.9208
-0.9821	0.8806	0.9633
-0.9541	0.8806	0.9633
-0.9156	0.8806	0.9758
-0.7188	0.8806	0.9808
-0.6340	0.8657	0.9842
-0.6179	0.8657	0.9842
-0.5882	0.8433	0.9842
-0.5625	0.8433	0.9875
-0.5417	0.8284	0.9917
-0.4521	0.8134	0.9917
-0.3044	0.7910	0.9933
0.0172	0.7537	0.9967
0.0531	0.7537	0.9967
0.3819	0.6866	0.9983
0.4360	**0.6866**	**0.9983**
0.5876	0.6194	0.9983
0.8071	0.5448	0.9992
0.856	**0.5448**	**0.9992**
1.0149	0.3582	1
1.0570	< 0.3582	1
1.1004	< 0.3582	1
1.3323	< 0.3582	1
1.3577	< 0.3582	1
1.4609	< 0.3582	1
1.6919	< 0.3582	1
1.7867	< 0.3582	1
2.6466	< 0.3582	1

Table 18: All 5 Study Queries (Q-Exp and Q-Comp)
This table lists all 5 queries used in this study. Q-Exp1 and Q-Exp2 are queries developed by librarians with extensive MEDLINE search experience. Q-Comp3, Q-Comp4, and Q-Comp5 are the computer-generated queries developed as part of this research. Chapters II and III describe the performance of these queries on the study dataset of 2000 citations. Chapter IV describes the performance of all five queries on a full year of MEDLINE publications.

Query Name	Query Detail
Query 1, Expert (Q-Exp1)	("drug interactions"[TIAB] NOT Medline[SB]) OR "drug interactions"[MeSH] OR drug interaction[Text Word]
Query 2, Expert (Q-Exp2)	"drug interactions"[MeSH] OR drug interactions[Text Word] AND ("Toxicity Tests"[MeSH] OR "Adverse Drug Reaction Reporting Systems"[MeSH] OR "Drug Hypersensitivity"[MeSH] OR "Drug Antagonism"[MeSH] OR "drugs, investigational"[MeSH] OR "Drug evaluation"[MeSH] OR "adverse effects"[sh] OR "toxicity"[sh] OR "poisoning"[Subheading] OR "chemically induced"[sh] OR "contraindications"[sh])
Query 3, Computer (Q-Comp3)	"drug interactions"[MeSH] OR ("humans"[MeSH] AND "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND "adverse effects"[sh] AND receiv* NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND "adverse effects"[sh] AND interact* NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH] NOT receiv*) OR ("humans"[MeSH] AND "adverse effects"[sh] NOT interact* NOT cell* NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH] NOT receiv*) OR ("humans"[MeSH] AND "pharmacology"[MeSH] NOT cell* NOT "adverse effects"[sh] NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH]) OR ("humans"[MeSH] AND receiv* NOT cell* NOT "pharmacology"[MeSH] NOT "drug synergism"[MeSH] NOT "pharmacokinetics"[MeSH] NOT "drug interactions"[MeSH] NOT "adverse effects"[sh]) OR ("pharmacology"[MeSH] AND interact* NOT cell* NOT "humans"[MeSH] NOT "drug interactions"[MeSH]) OR (interact* AND "drug synergism"[MeSH] NOT "pharmacology"[MeSH] NOT cell* NOT "humans"[MeSH] NOT "drug interactions"[MeSH])
Query 4, Computer (Q-Comp4)	("drug interactions"[MeSH] AND "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND interact* NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND receiv* NOT interact* NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND "humans"[MeSH] NOT receiv* NOT interact* NOT "pharmacokinetics"[MeSH])
Query 5, Computer (Q-Comp5)	("drug interactions"[MeSH] AND "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND interact* AND "pharmacology"[MeSH] NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND interact* AND "adverse effects"[sh] NOT "pharmacology"[MeSH] NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND receiv* NOT interact* NOT "pharmacokinetics"[MeSH]) OR ("drug interactions"[MeSH] AND "humans"[MeSH] AND "pharmacology"[MeSH] NOT receiv* NOT interact* NOT "pharmacokinetics"[MeSH])

APPENDIX B

SUPPLEMENTARY FIGURES

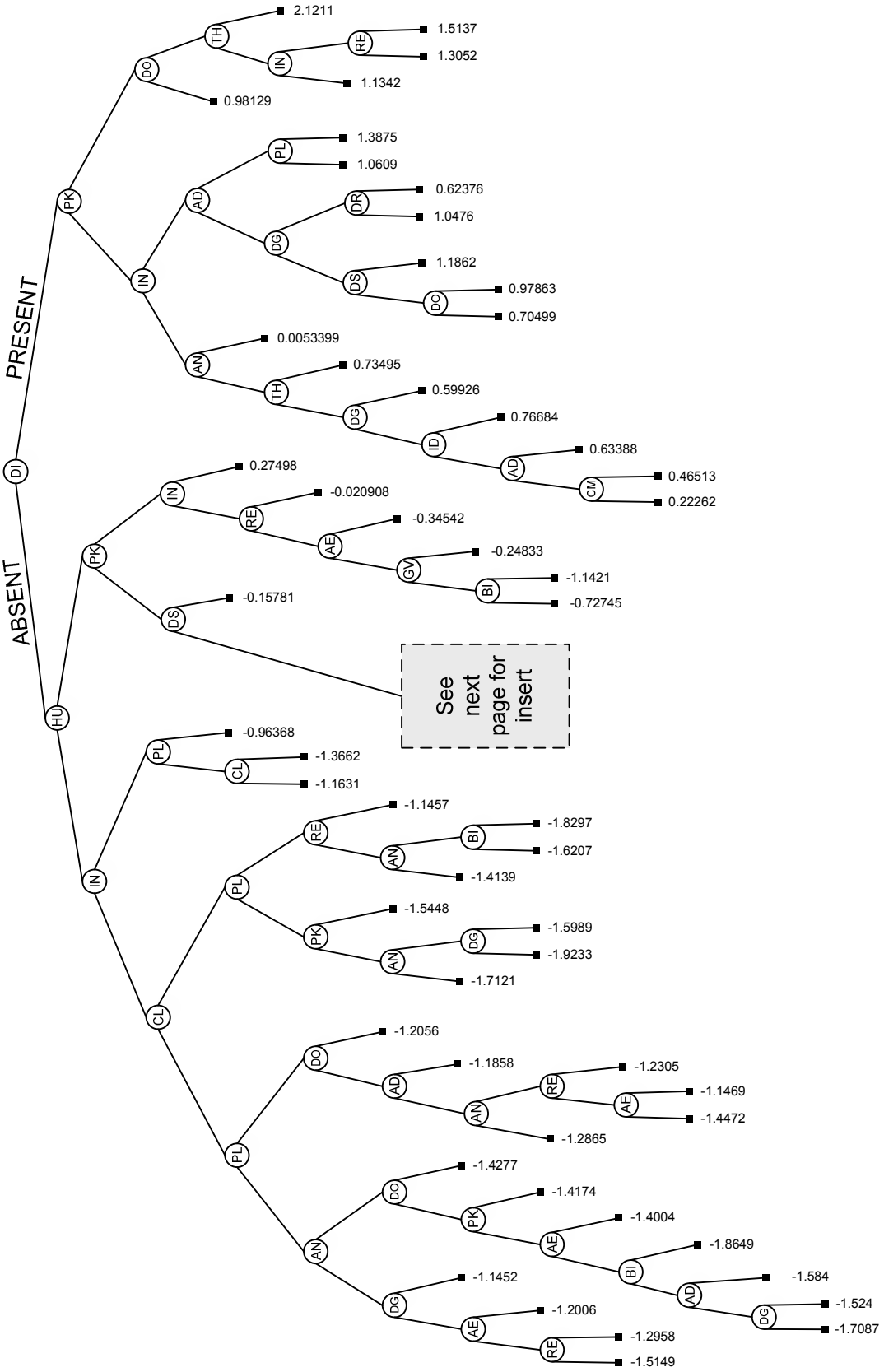

Figure 9: Tree MB
This image depicts the decision tree. Tree MB and its outputs. Nodes (circled) represent tests for text word or MeSH terms in the document. The figure legend is found on the next page. The portion of Tree MB not depicted above (gray box) is pictured in Figure 10.

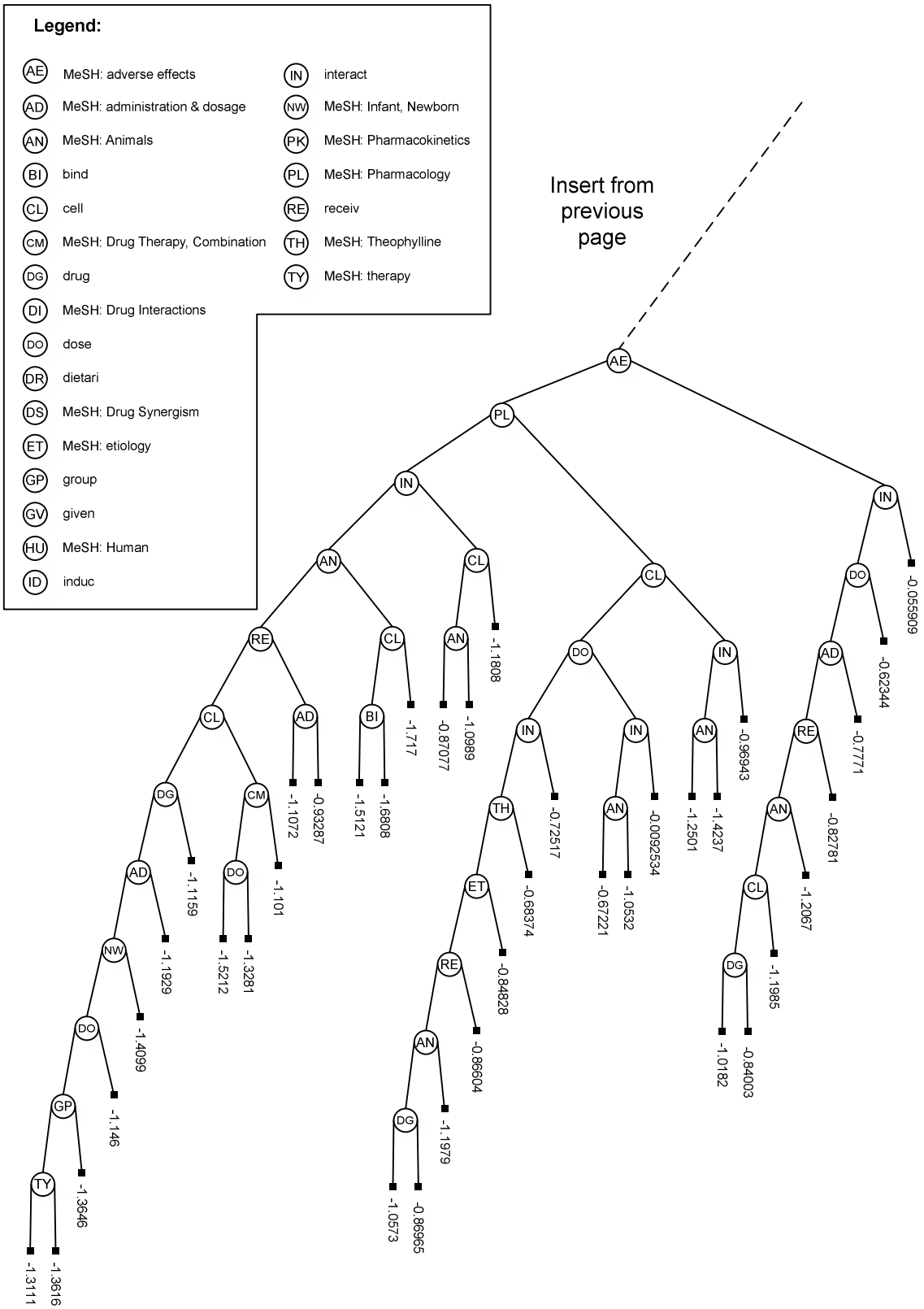

Figure 10: Tree MB Insert

Figure 10 fills in the area marked by a gray box in Figure 9's depiction of Tree MB. The top-most AE node in Figure 10 is the left child of the DS node preceding the gray box in Figure 9.

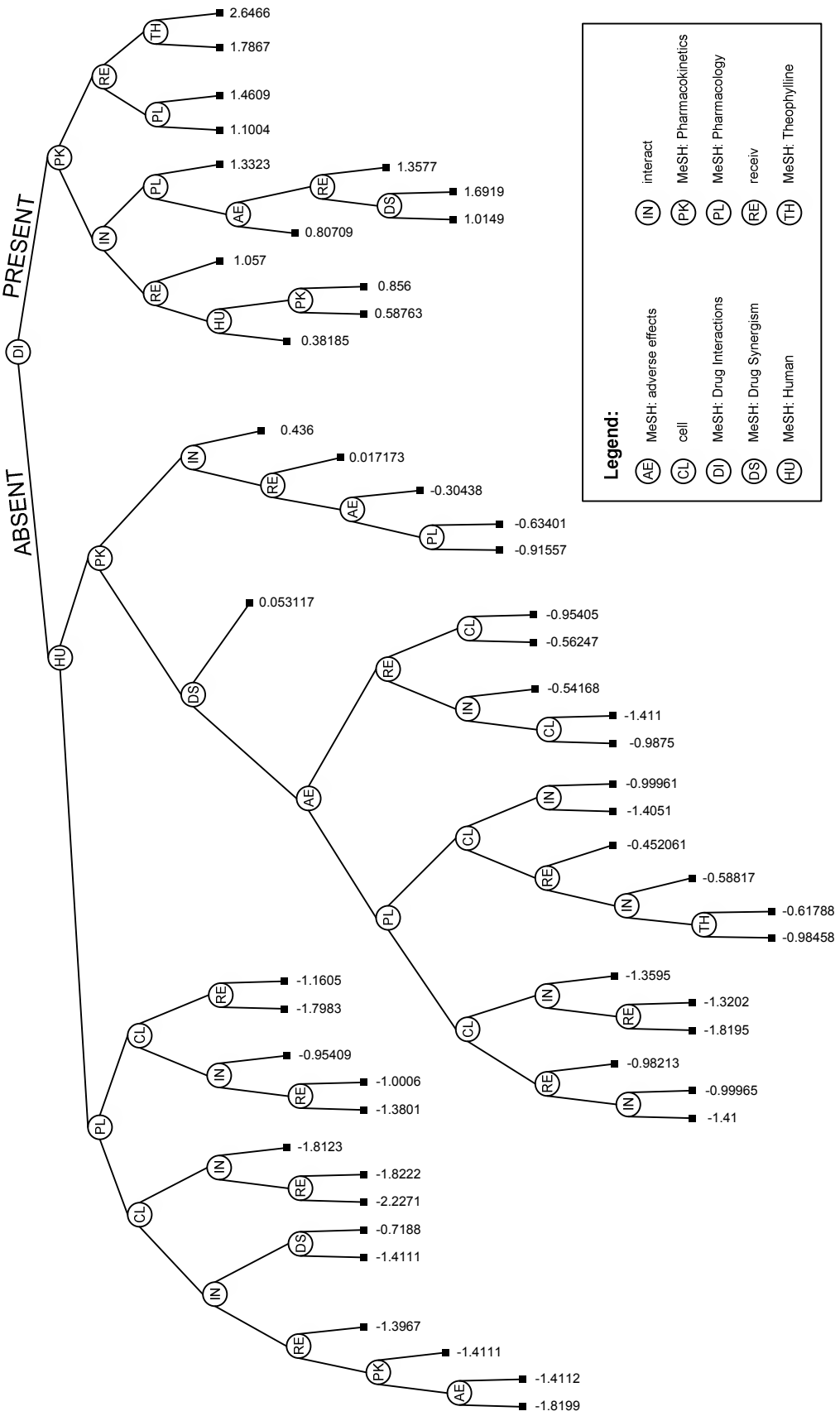

Figure 11: Tree PC
This image depicts the decision tree Tree PC and its outputs. Nodes (circled) represent tests for text word or MeSH terms in the document. The ten test words are listed in the legend. Right tree branches indicate a term is present in the document, left branches indicate a term is absent.

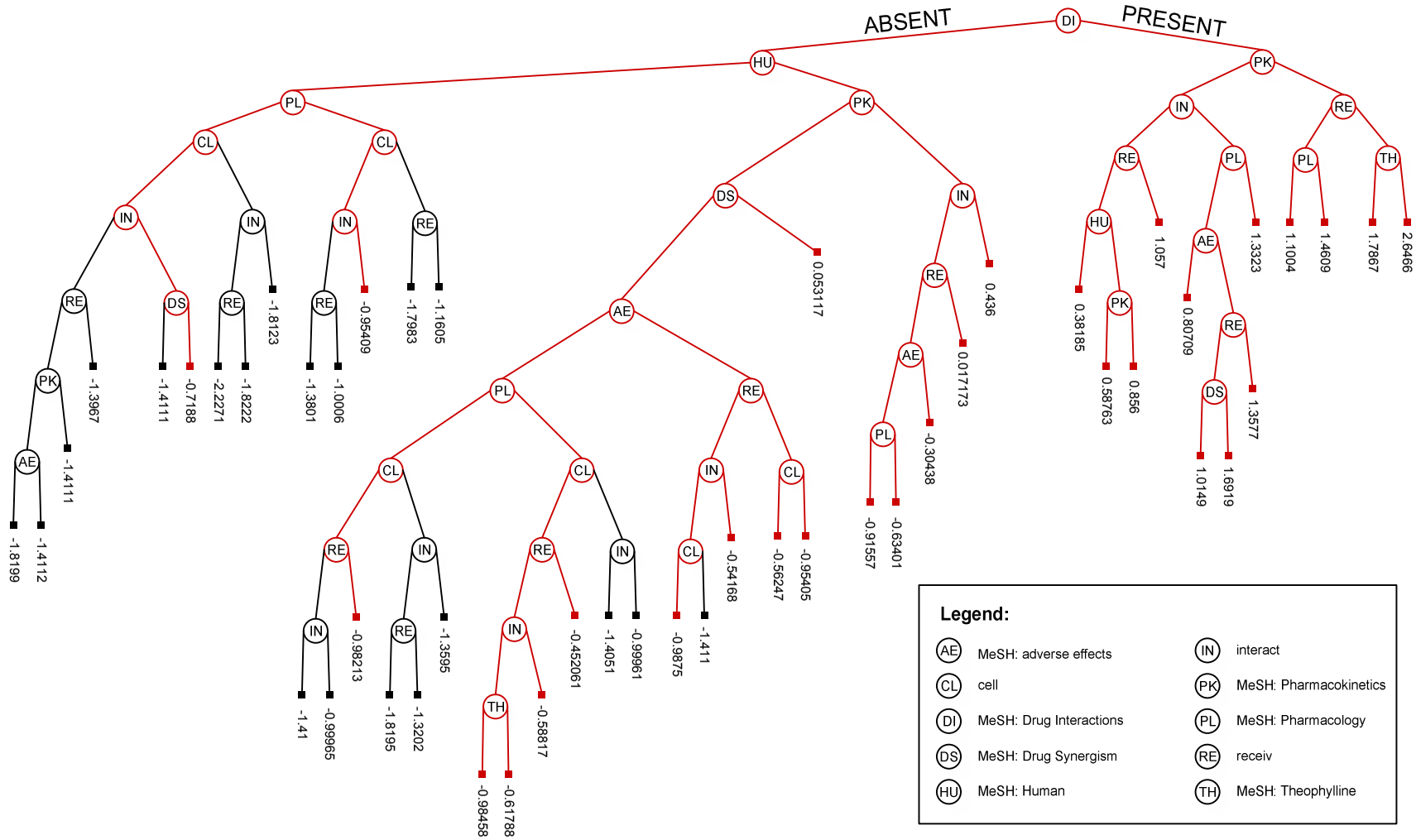

Figure 12: Tree PC with DT-3 Overlay

This image depicts the decision tree Tree PC and its outputs with the paths of DT-3 overlaid in red. DT-3 is formed of all branches terminating in leaf nodes with values greater than or equal to -0.9875.

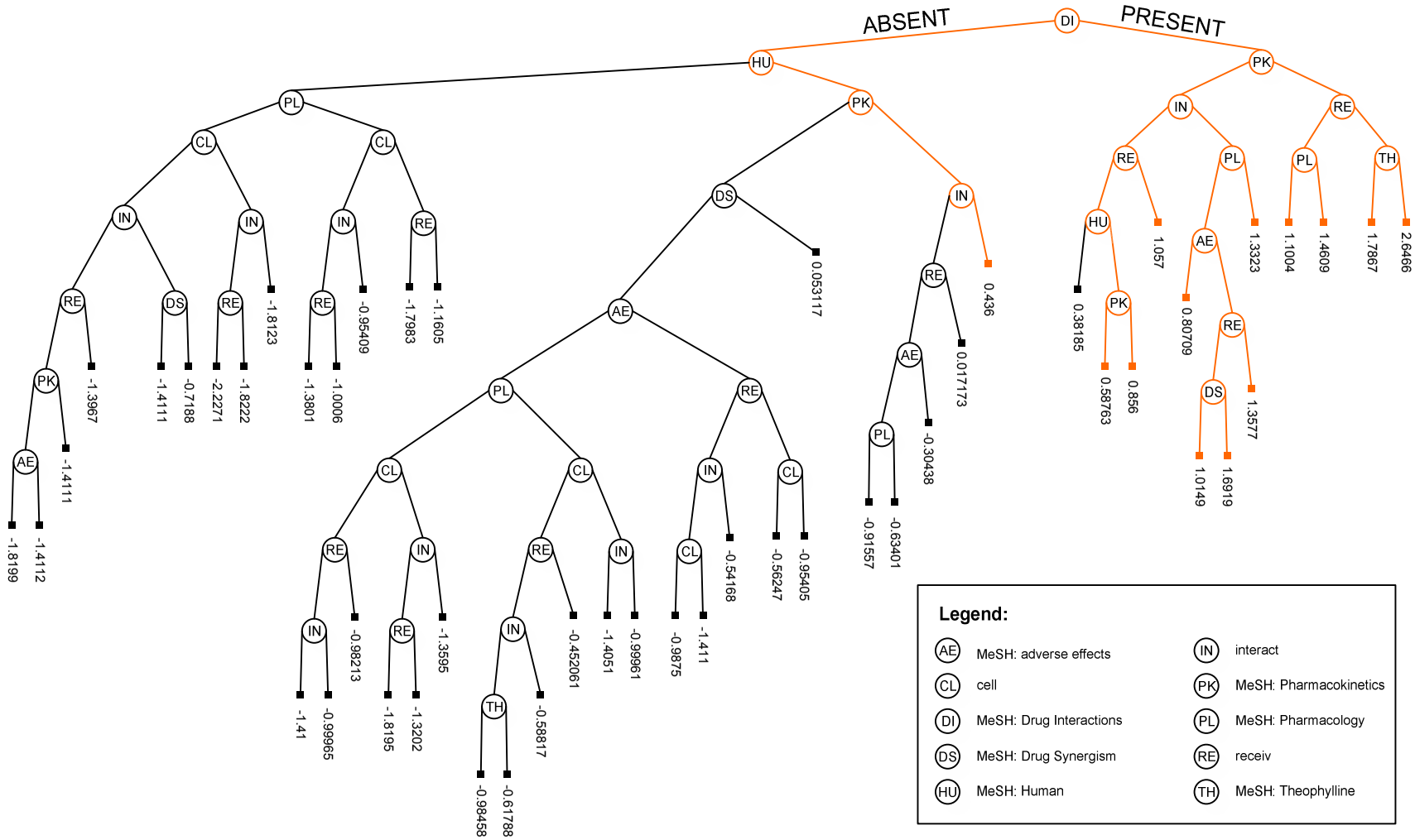

Figure 13: Tree PC with DT-4 Overlay

This image depicts the decision tree Tree PC and its outputs with the paths of DT-4 overlaid in orange. DT-4 is formed of all branches terminating in leaf nodes with values greater than or equal to 0.436.

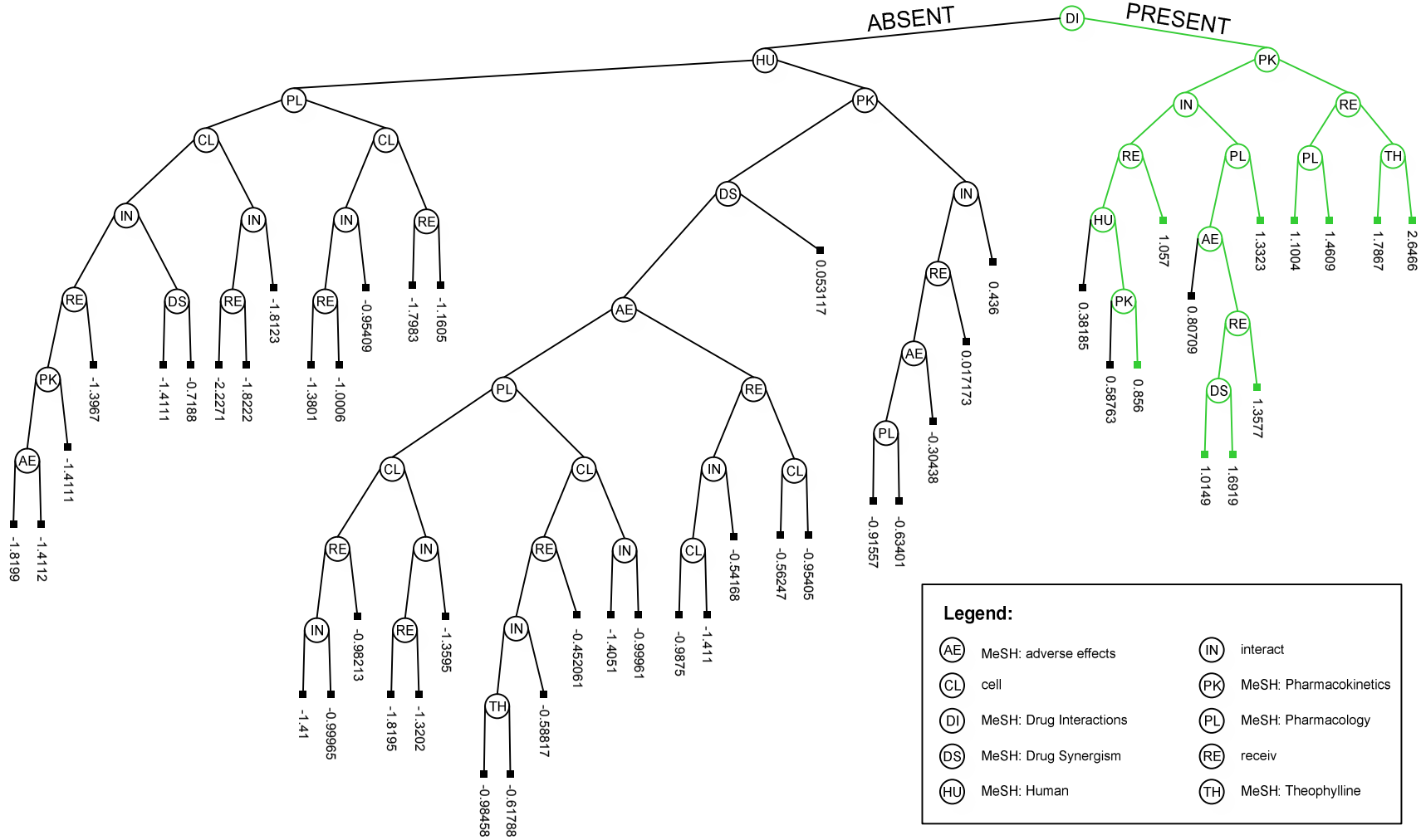

Figure 14: Tree PC with DT-5 Overlay

This image depicts the decision tree Tree PC and its outputs with the paths of DT-5 overlaid in green. DT-5 is formed of all branches terminating in leaf nodes with values greater than or equal to 0.856.

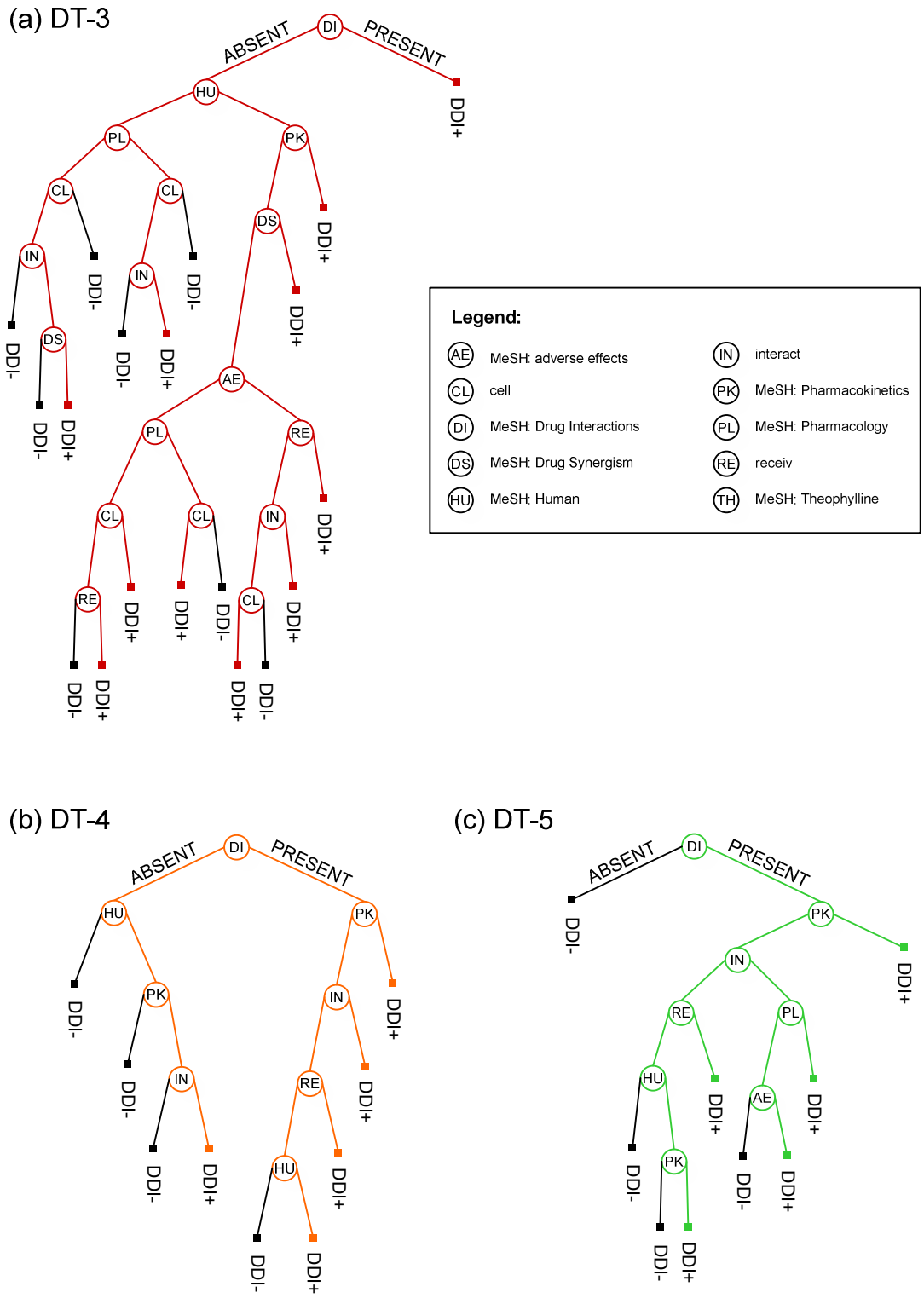

Figure 15: Pruned Versions of Binary DT-3, DT-4, and DT-5
Figures (a) through (c) are pruned, binary versions of the decision trees in Figures 9-11. Nodes (circled) represent present/absent tests for stemmed text words and MeSH terms. The left branch symbolizes the term is absent in the document, the right branch indicates the term is present in the text. Leaf nodes assign DDI+ or DDI- classifications to documents.

BIBLIOGRAPHY

1. Committee on Quality of Health Care in America, I.o.M., *To Err is Human: Building a Safer Health System*, ed. J.M.C. Linda T. Kohn, and Molla S. Donaldson. 2000: The National Academies Press.

2. Peddicord, T.E., et al., *A casting call from industry: reel in and retain appropriate information, release the rest.* Pharmacotherapy, 2002. **22**(7): p. 934-7; discussion 937-8.

3. Bates, D.W., et al., *Incidence of adverse drug events and potential adverse drug events. Implications for prevention. ADE Prevention Study Group.* Jama, 1995. **274**(1): p. 29-34.

4. Ghandi, T.K., et al., *Adverse Drug Events in Ambulatory Care.* N Engl J Med, 2003. **348**(16): p. 1556-1564.

5. Lazarou, J., B. Pomeranz, and P. Corey, *Incidence of Adverse Drug Reactions in Hospitalized Patients: A meta-analysis of Prospective Studies.* JAMA, 1998. **279**(15): p. 1200-1205.

6. Morimoto, T., et al., *Adverse drug events and medication errors: detection and classification methods.* Qual Saf Health Care, 2004. **13**: p. 306-314.

7. Classen, D.C., et al., *Adverse drug events in hospitalized patients. Excess length of stay, extra costs, and attributable mortality.* JAMA, 1997. **277**(4): p. 301-6.

8. Gurwitz, J.H., et al., *Incidence and preventability of adverse drug events among older persons in the ambulatory setting.* JAMA, 2003. **289**(9): p. 1107-16.

9. Sullivan, J.E. and J.J. Buchino, *Medication errors in pediatrics--the octopus evading defeat.* J Surg Oncol, 2004. **88**(3): p. 182-8.

10. Kelly, W.N., *Potential risks and prevention, Part 4: Reports of significant adverse drug events.* Am J Health Syst Pharm, 2001. **58**(15): p. 1406-12.

11. Gurwitz, J.H., et al., *Incidence and preventability of adverse drug events in nursing homes.* Am J Med, 2000. **109**(2): p. 87-94.

12. Johnson, J.A. and J.L. Bootman, *Drug-related morbidity and mortality. A cost-of-illness model.* Arch Intern Med, 1995. **155**(18): p. 1949-56.

13. Bates, D.W., et al., *The costs of adverse drug events in hospitalized patients. Adverse Drug Events Prevention Study Group.* JAMA, 1997. **277**(4): p. 307-11.

14. Field, T.S., et al., *Strategies for detecting adverse drug events among older persons in the ambulatory setting.* J Am Med Inform Assoc, 2004. **11**(6): p. 492-8.

15. McDonnell, P.J. and M.R. Jacobs, *Hospital admissions resulting from preventable adverse drug reactions.* Ann Pharmacother, 2002. **36**(9): p. 1331-6.

16. *"Drug interaction."* in *The American Heritage Stedman's Medical Dictionary.* 2002, Houghton Mifflin Company.

17. *ASHP Patient Concerns National Survey Research Report, 1999*. 1999, American Society of Health Systems.

18. Raschetti, R., et al., *Suspected adverse drug events requiring emergency department visits or hospital admissions.* Eur J Clin Pharmacol, 1999. **54**(12): p. 959-63.

19. Stockley, I.H., ed. *Stockley's Drug Interactions*. 6 ed. 2002, Pharmaceutical Press: London.

20. Grymonpre, R.E., et al., *Drug-associated hospital admissions in older medical patients.* J Am Geriatr Soc, 1988. **36**(12): p. 1092-8.

21. Jankel, C.A. and S.M. Speedie, *Detecting drug interactions: a review of the literature.* Dicp, 1990. **24**(10): p. 982-9.

22. Hamilton, R.A., L.L. Briceland, and M.H. Andritz, *Frequency of hospitalization after exposure to known drug-drug interactions in a Medicaid population.* Pharmacotherapy, 1998. **18**(5): p. 1112-20.

23. Langdorf, M.I., et al., *Physician versus computer knowledge of potential drug interactions in the emergency department.* Acad Emerg Med, 2000. **7**(11): p. 1321-9.

24. Lyles, A., et al., *When Warnings Are Not Enough: Primary Prevention Through Drug Use Review.* Health Affiars, 1998. **Sept/Oct**(5): p. 175-183.

25. Glassman, P.A., et al., *Improving Recognition of Drug Interactions: Benefits and Barriers to Using Automated Drug Alerts.* Med Care, 2002. **40**(12): p. 1161-1171.

26. Bergk, V., et al., *Requirements for a successful implementation of drug interaction information systems in general practice: results of a questionnaire survey in Germany.* Eur J Clin Pharmacol, 2004. **60**(8): p. 595-602.

27. Kelly, W.N., *Drug interaction monitoring: Good, bad, and ugly.* Drug Topics, 2003. **147**: p. 41.

28. Headden, S., et al., *Danger at the Drugstore: Too many pharmacists fail to protect consumers against potentially hazardous interactions of prescription drugs*, in *U.S. News & World Report*. 1996. p. 6.

29. eHealth Initiative, *Electronic Prescribing: Toward Maximum Value and Rapid Adoption*. 2004

30. Teich, J.M., et al., *Effects of computerized physician order entry on prescribing practices.* Arch Intern Med, 2000. **160**(18): p. 2741-7.

31. Bates, D.W., et al., *Effect of computerized physician order entry and a team intervention on prevention of serious medication errors.* JAMA, 1998. **280**(15): p. 1311-6.

32. Sommer, C., *Clinical Pharmacist, First DataBank*. [personal communication] May 12, 2004

33. Osheroff, J., *Chief Clinical Informatics Officer, Thompson Micromedex*. [personal communication] Mar 7, 2005

34. Robinson, G., *VP of Knowledge Base Development, First DataBank*. [personal communication] April 23, 2004

35. *Cerner Multum Glossary. Interactions: Drug-Drug and Drug-Food.* 2005 [cited 2005 September 12]; Available from: http://www.multum.com/Glossary.htm

36. Fulda, T.R., et al., *Disagreement among drug compendia on inclusion and ratings of drug-drug interactions.* Curr Ther Res Clin Exp, 2000. **61**: p. 540-548.

37. Jankel, C.A. and B.C. Martin, *Evaluation of six computerized drug interaction screening programs.* Am J Hosp Pharm, 1992. **49**(6): p. 1430-5.

38. Hazlet, T.K., et al., *Performance of community pharmacy drug interaction software.* J Am Pharm Assoc (Wash), 2001. **41**(2): p. 200-4.

39. Smith, W.D., et al., *Evaluation of drug interaction software to identify alerts for transplant medications.* Ann Pharmacother, 2005. **39**(1): p. 45-50.

40. Clauson, K.A., et al., *Evaluation of drug information databases for personal digital assistants.* Am J Health Syst Pharm, 2004. **61**(10): p. 1015-24.

41. Enders, S.J., J.M. Enders, and S.G. Holstad, *Drug-information software for Palm operating system personal digital assistants: breadth, clinical dependability, and ease of use.* Pharmacotherapy, 2002. **22**(8): p. 1036-40.

42. Abarca, J., et al., *Concordance of severity ratings provided in four drug interaction compendia.* J Am Pharm Assoc (Wash DC), 2004. **44**(2): p. 136-41.

43. Cavuto, N.J., R.L. Woosley, and M. Sale, *Pharmacies and prevention of potentially fatal drug interactions.* JAMA, 1996. **275**(14): p. 1086-7.

44. Weingart, S.N., et al., *Physician's Decisions to Override Computerized Drug Alerts in Primary Care.* Arch Intern Med, 2003. **163**: p. 2625-2631.

45. Spina, J.R., et al., *Clinical relevance of automated drug alerts from the perspective of medical providers.* Am J Med Qual, 2005. **20**(1): p. 7-14.

46. Chui, M.A. and M.T. Rupp, *Evaluation of online prospective DUR programs in community pharmacy practice.* J Manag Care Pharm, 2000. **6**: p. 27-32.

47. Payne, T.H., et al., *Characteristics and override rates of order checks in a practitioner order entry system.* Proc AMIA Symp, 2002: p. 602-6.

48. Hansten, P.D., *Drug interaction management.* Pharm World Sci, 2003. **25**(3): p. 94-7.

49. Hsieh, T.C., et al., *Characteristics and consequences of drug allergy alert overrides in a computerized physician order entry system.* J Am Med Inform Assoc, 2004. **11**(6): p. 482-91.

50. Limon, L. *Drug Interactions: Decreasing the Risk*. in *American Pharmaceutical Association Annual Meeting, 2000*. 2000.

51. Malone, D.C., et al., *Identification of serious drug-drug interactions: results of the partnership to prevent drug-drug interactions.* J Am Pharm Assoc (Wash DC), 2004. **44**(2): p. 142-51.

52. McMullin, S.T., et al., *Impact of a Web-Based Clinical Information System on Cisapride Drug Interactions and Patient Safety.* Arch Intern Med, 1999. **159**: p. 2077-2082.

53. Pedersen, D., *Drug-Drug Interaction Initiative to provide pharmacists with reliable drug interaction messaging*. 2004, Pharmaceutical News.

54. Gardner, R.M. and R.S. Evans, *Using computer technology to detect, measure, and prevent adverse drug events.* J Am Med Inform Assoc, 2004. **11**(6): p. 535-6.

55. NLM. *Fact Sheet: PubMed: MEDLINE Retrieval on the World Wide Web*. 2002 March 2005 [cited 2005 April 12, 2005]; Available from: http://www.nlm.nih.gov/pubs/factsheets/pubmed.html

56. Barillot, M.J., B. Sarrut, and C.G. Doreau, *Evaluation of drug interaction document citation in nine on-line bibliographic databases.* Ann Pharmacother, 1997. **31**(1): p. 45-9.

57. Hunink, M. and P. Glasziou, *Decision making in health and medicine*. 2001, Cambridge: Cambridge University Press.

58. NLM. *Fact Sheet: Medical Subject Headings (MeSH).* 2005 27 May 2005 [cited 2005 14 June]; Available from: http://www.nlm.nih.gov/pubs/factsheets/mesh.html

59. Bronander, K.A., et al., *Boolean search experience and abilities of medical students and practicing physicians.* Teach Learn Med, 2004. **16**(3): p. 284-9.

60. Slingluff, D., Y. Lev, and A. Eisan, *An end user search service in an academic health sciences library.* Med Ref Serv Q, 1985. **4**(1): p. 11-21.

61. Humphrey, S.M., *File maintenance of MeSH headings in MEDLINE.* J Am Soc Inf Sci, 1984. **35**(1): p. 34-44.

62. Hersh, W.R. and D.H. Hickam, *How well do physicians use electronic information retrieval systems? A framework for investigation and systematic review.* JAMA, 1998. **280**(15): p. 1347-52.

63. McKibbon, K.A., et al., *How good are clinical MEDLINE searches? A comparative study of clinical end-user and librarian searches.* Comput Biomed Res, 1990. **23**(6): p. 583-93.

64. Haynes, R.B., et al., *A program to enhance clinical use of MEDLINE. A randomized controlled trial.* Online J Curr Clin Trials, 1993. **Doc No 56**: p. [4005 words; 39 paragraphs].

65. Haynes, R.B., et al., *Developing optimal search strategies for detecting clinically sound studies in MEDLINE.* J Am Med Inform Assoc, 1994. **1**(6): p. 447-58.

66. Wilczynski, N.L. and R.B. Haynes, *Optimal search strategies for detecting clinically sound prognostic studies in EMBASE: an analytic survey.* J Am Med Inform Assoc, 2005. **12**(4): p. 481-5.

67. Wilczynski, N.L., D. Morgan, and R.B. Haynes, *An overview of the design and methods for retrieving high-quality studies for clinical care.* BMC Med Inform Decis Mak, 2005. **5**: p. 20.

68. Wilczynski, N.L., et al., *Optimal search strategies for detecting health services research studies in MEDLINE.* Cmaj, 2004. **171**(10): p. 1179-85.

69. Nwosu, C.R., K.S. Khan, and P.F. Chien, *A two-term MEDLINE search strategy for identifying randomized trials in obstetrics and gynecology.* Obstet Gynecol, 1998. **91**(4): p. 618-22.

70. Robinson, K.A. and K. Dickersin, *Development of a highly sensitive search strategy for the retrieval of reports of controlled trials using PubMed.* Int J Epidemiol, 2002. **31**(1): p. 150-3.

71. Jenuwine, E.S. and J.A. Floyd, *Comparison of Medical Subject Headings and text-word searches in MEDLINE to retrieve studies on sleep in healthy individuals.* J Med Libr Assoc, 2004. **92**(3): p. 349-53.

72. Bachmann, L.M., et al., *Identifying Diagnostic Studies in MEDLINE: Reducing the Number Needed to Read.* J Am Med Inform Assoc, 2002. **9**(6): p. 653-658.

73. Bartling, W.C., T.K. Schleyer, and S. Visweswaran, *Retrieval and classification of dental research articles.* Adv Dent Res, 2003. **17**: p. 115-20.

74. Suomela, B.P. and M.A. Andrade, *Ranking the whole MEDLINE database according to a large training set using text indexing.* BMC Bioinformatics, 2005. **6**(1): p. 75.

75. Rindflesch, T.C., L. Hunter, and A.R. Aronson, *Mining molecular binding terminology from biomedical text.* Proc AMIA Symp, 1999: p. 127-31.

76. McCray, A.T., et al., *UMLS knowledge for biomedical language processing.* Bull Med Libr Assoc, 1993. **81**(2): p. 184-94.

77. Zou, Q., et al., *IndexFinder: A Method of Extracting Key Concepts from Clinical Texts for Indexing.* AMIA Annu Symp Proc, 2003. **2003**(2003): p. 763-767.

78. NLM. *Fact Sheet: Unified Medical Language System.* 2003 March 2004 [cited 2005 August 14, 2005]; Available from: http://www.nlm.nih.gov/pubs/factsheets/umls.html

79. Bodenreider, O. *Using UMLS Semantics for Classification Purposes*. in *AMIA Annual Symposium 2000*. 2000: AMIA, Inc.

80. Bachmann, K.A. and J.D. Lewis, *Predicting inhibitory drug-drug interactions and evaluating drug interaction reports using inhibition constants.* Ann Pharmacother, 2005. **39**(6): p. 1064-72.

81. Blaschke, C., et al., *Automatic extraction of biological information from scientific text: protein-protein interactions.* Proc Int Conf Intell Syst Mol Biol, 1999: p. 60-7.

82. Rubin, D.L., et al., *A statistical approach to scanning the biomedical literature for pharmacogenetics knowledge.* J Am Med Inform Assoc, 2005. **12**(2): p. 121-9.

83. Libbus, B. and T.C. Rindflesch, *NLP-based information extraction for managing the molecular biology literature.* Proc AMIA Symp, 2002: p. 445-9.

84. Bate, A., et al., *A Bayesian neural network method for adverse drug reaction signal generation.* Eur J Clin Pharmacol, 1998. **54**(4): p. 315-21.

85. Wilczynski, N.L. and R.B. Haynes, *Robustness of empirical search strategies for clinical content in MEDLINE.* Proc AMIA Symp, 2002: p. 904-8.

86. Rindflesch, T.C. and M. Fiszman, *The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text.* J Biomed Inform, 2003. **36**(6): p. 462-77.

87. Berrios, D.C., R.J. Cucina, and L.M. Fagan, *Methods for Semi-automated Indexing for High Precision Information Retrieval.* J Am Med Inform Assoc, 2002. **9**(6): p. 637-652.

88. Hope, C., et al., *A tiered approach is more cost effective than traditional pharmacist-based review for classifying computer-detected signals as adverse drug events.* J Biomed Inform, 2003. **36**(1-2): p. 92-8.

89. Russell, S.J. and P. Norvig, *Artificial Intelligence: A Modern Approach*. 2 ed. 2003, Upper Saddle River, NJ: Pearson Education, Inc.

90. Gunn, S.R. *Support Vector Machines for Classification and Regression*. 1998 [cited 2005 3 September]; Available from: http://www.ecs.soton.ac.uk/~srg/publications/pdf/SVM.pdf

91. Hsu, C.-W., C.-C. Chang, and C.-J. Lin. *A Practical Guide to Support Vector Classification*. 2005 [cited 2005 12 Jun]; Available from: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.

92. Mitchell, T.M., *Machine Learning*. 1997, Boston: WCB/McGraw-Hill.

93. Guyon, I. and A. Elisseeff, *An Introduction to Variable and Feature Selection.* Journal of Machine Learning Research, 2003. **3**(March 2003): p. 1157 - 1182

94. Yang, Y. and J.O. Pedersen. *A Comparative Study on Feature Selection in Text Categorization*. in *ICML-97, 14th International Conference on Machine Learning*. 1997.

95. Burges, C., *A Tutorial on Support Vector Machines for Pattern Recognition.* Data Mining and Knowledge Discovery, 1998. **2**(2): p. 121 - 167.

96. Joachims, T. *Text Categorization with Support Vector Machines: Learning with Many Relevant Features*. in *ECML-98, 10th European Conference on Machine Learning*. 1998: Springer Verlag, Heidelberg, DE.

97. Aphinyanaphongs, Y., et al., *Text categorization models for high-quality article retrieval in internal medicine.* J Am Med Inform Assoc, 2005. **12**(2): p. 207-16.

98. Joachims, T., *Learning to classify text using support vector machines*. Kluwer international series in engineering and computer science ; SECS 668. 2002, Boston: Kluwer Academic Publishers. xvi, 205 p.

99. Joachims, T., *Learning to Classify Text Using Support Vector Machines : Methods, Theory and Algorithms*. 2002: Springer.

100. Markowetz, F., *Support Vector Machines in Bioinformatics*, in *Department of Mathematics*. 2003, Ruprecht-Karls Universitat Heidelberg: Heidelberg. p. 100.

101. Núñez, H., C. Angulo, and A. Català. *Rule extraction from support vector machines*. in *ESANN'2002* 2002. Belgium.

102. Burges, C. *Simplified Support Vector Decision Rules*. in *13th International Conference on Machine Learning*. 1996.

103. Aphinyanaphongs, Y. and C.F. Aliferis, *Learning boolean queries for article quality filtering.* Medinfo, 2004. **2004**: p. 263-7.

104. Potts, A.L., et al., *Computerized physician order entry and medication errors in a pediatric critical care unit.* Pediatrics, 2004. **113**(1 Pt 1): p. 59-63.

105. Kupferberg, N. and L. Jones Hartel, *Evaluation of five full-text drug databases by pharmacy students, faculty, and librarians: do the groups agree?* J Med Libr Assoc, 2004. **92**(1): p. 66-71.

106. NLM. *EFetch Overview*. 2005 27 Jul 2005 [cited 2005 12 Jan]; Available from: http://eutils.ncbi.nlm.nih.gov/entrez/query/static/efetch_help.html

107. Bjerrum, L., et al., *Exposure to potential drug interactions in primary health care.* Scand J Prim Health Care, 2003. **21**(3): p. 153-8.

108. NLM. *NCBI Help Manual: Table 7. Stopwords*. 2004 [cited 2004 June 16]; Available from: http://www.ncbi.nlm.nih.gov/books/bv.fcgi?rid=helppubmed.table.pubmedhelp.T43

109. Bates, D.W., *Frequency, consequences and prevention of adverse drug events.* J Qual Clin Pract, 1999. **19**(1): p. 13-7.

110. Porter, M., *An algorithm for suffix stripping.* Program, 1980. **14**(3): p. 130-137.

111. Chang, C.-C. and C.-J. Lin. *LIBSVM: a library for Support Vector Machines*. 2004 [cited; Available from: http://www.csie.ntu.edu.tw/~cjlin/libsvm

112. Aliferis, C.F., I. Tsamardinos, and A. Statnikov, *HITON: a novel Markov Blanket algorithm for optimal variable selection.* AMIA Annu Symp Proc, 2003: p. 21-5.

113. Blum, A.L. and P. Langley, *Selection of Relevant Features and Examples in Machine Learning* Artificial Intelligence, 1997. **97**(1-2): p. 245-271.

114. Ding, J., et al., *Mining MEDLINE: abstracts, sentences, or phrases?* Pac Symp Biocomput, 2002: p. 326-37.

115. Hahn, U., M. Romacker, and S. Schulz, *Creating knowledge repositories from biomedical reports: the MEDSYNDIKATE text mining system.* Pac Symp Biocomput, 2002: p. 338-49.

116. Bodenreider, O., et al., *Beyond synonymy: exploiting the UMLS semantics in mapping vocabularies.* Proc AMIA Symp, 1998: p. 815-9.

117. Flake, G., et al. *Extracting Query Modifications from Nonlinear {SVMs}*. in *International World Wide Web Conference*. 2002.

118. MATLAB. *Statistics Toolbox v5.0.2*. 2005 [cited 2005 4 Apr]; Available from: http://www.mathworks.com/products/statistics/

119. Aronson, A.R. and T.C. Rindflesch, *Query expansion using the UMLS Metathesaurus.* Proc AMIA Annu Fall Symp, 1997: p. 485-9.

120. Hersh, W., S. Price, and L. Donohoe, *Assessing thesaurus-based query expansion using the UMLS Metathesaurus.* Proc AMIA Symp, 2000: p. 344-8.

121. Aphinyanaphongs, Y. and C.F. Aliferis, *Text categorization models for retrieval of high quality articles in internal medicine.* AMIA Annu Symp Proc, 2003: p. 31-5.

122. Balas, E.A., et al., *In search of controlled evidence for health care quality improvement.* J Med Syst, 1997. **21**(1): p. 21-32.

123. Backus, J.E., S. Davidson, and R. Rada, *Searching for patterns in the MeSH vocabulary.* Bull Med Libr Assoc, 1987. **75**(3): p. 221-7.

124. Lam, M.V., G.M. McCart, and C. Tsourounis, *An Assessment of Free, Online Drug-Drug Interaction Screening Programs (DSPs).* Hosp Pharm, 2003. **38**(7): p. 662-668.

125. Chang, C.-C. and C.-J. Lin. *LIBSVM: a library for Support Vector Machines*. 2001 [cited 2005 2 May]; Available from: http://www.csie.ntu.edu.tw/~cjlin/libsvm

126. Wu, T.-F., C.-J. Lin, and R.C. Weng, *Probability estimates for multi-class classification by pairwise coupling.* Journal of Machine Learning Research, 2004. **5**: p. 975-1005.