A Verb-Based Approach to Mining Adverse Drug Side-Effects from Online Health Care Forums

by

Jack Hester

Thesis Adviser

Dr. Mark Risjord

Institute of the Liberal Arts

Interdisciplinary Studies in Societies and Cultures

Academic Adviser

Dr. Peter Wakefield

Additional Faculty Adviser

Dr. Roberto Franzosi

2019

A Verb-Based Approach to Mining Adverse Drug Side-Effects from Online Health Care Forums

by

Jack Hester

Thesis Adviser

Dr. Mark Risjord

An abstract of

a thesis submitted to the Faculty of the Emory College of Arts and Sciences

of Emory University in partial fulfillment

of the requirements of the degree of

Bachelor of Arts in Interdisciplinary Studies in Societies and Cultures

2019

Abstract

Adverse drug side-effects (ADEs) cause countless deaths and injuries each year, and they are a large contributor to ineffective and wasteful spending. These harms can be reduced by pharmacovigilance, but the FDA's (Food and Drug Administration's) current post-clinical-trial monitoring methods are somewhat lacking. Continuous analysis of online health care forums, among other text sources, can aid in the monitoring process. Previous work has demonstrated the potential for successful extraction of ADEs from online sources, but the methods have relied on large dictionaries. This paper proposes opportunities for moving away from those dictionary-reliant methods. The proposed approaches are evaluated by comparison of extracted ADEs to drug inserts, the original forum posts, and the FEARS (FDA Adverse Event Reporting System) database. Extraction that relies on a verbs-first approach without the necessity for a side-effect dictionary is shown to be effective. The methods provided are promising and, with future fine-tuning, may well be able to outperform the current dictionary-reliant ADE extraction routines.

A Verb-Based Approach to Mining Adverse Drug Side-Effects from Online Health Care Forums

by

Jack Hester

Thesis Adviser

Dr. Mark Risjord

A thesis submitted to the Faculty of the Emory College of Arts and Sciences

of Emory University in partial fulfillment

of the requirements of the degree of

Bachelor of Arts in Interdisciplinary Studies in Societies and Cultures

Institute of the Liberal Arts

Emory University

2019

Acknowledgements

Thank you to my parents who have supported me through everything.

Thank you to Dr. Risjord, my adviser in all of my IDS endeavors and beyond.

Thank you to Dr. Franzosi, who first exposed me to computational linguistics and continues to be a mentor and has provided me with countless opportunities to work with him.

Thank you to Dr. Wakefield and other members of the ILA who have guided and encouraged me throughout my college years.

Thank you to my friends who have kept me sane and been kind to me over the past few years: Cole, Kristin, Robert, Peter, Davis, Stephen, Sam, Luis, Tomo, Jonny, Izzie, and Marshall (among others).

To those hurt or killed by side effects...

Contents

List of Figures

List of Tables

Nomenclature and Acronyms

ADE	Adverse Drug Side-Effect
CoNLL	Coreference on Natural Language Learning
DEPREL	Dependency relationship
FEARS	FDA Adverse Event Reporting System
IDF	Inverse Document Frequency
ML	Machine learning
NAD	Newer analyzed drug
NER	Named Entity Recognition
NLP	Natural Language Processing
OAD	Older analyzed drug
POSTAG	Part of speech tag
SVM	Support Vector Machine
TF	Term Frequency
TF-IDF	Term Frequency-Inverse Document Frequency

1 Introduction

Prescription pharmaceutical drugs are, in the most general sense, chemical substances given to patients in order to treat or even cure medical conditions. While these drugs typically do more good than harm, there are often undesired and unintended negative side-effects. These adverse side-effects can range from mildly inconveniencing to deadly. The problem is so serious and widespread that an estimated 100,000 people die each year in the U.S. from adverse drug effects.[1, 2] Adverse side-effects are also a major contributor to an estimated 20-30% of U.S. health care spending on wasteful treatments and adverse events.[3]

Beyond the severity and cost of many side-effects, reducing negative patient experiences with prescription drugs—a form of harm reduction—is a worthwhile avenue to pursue in health care.

Clinical trials cannot catch every adverse side-effect, so constant side-effect monitoring is essential. Even so, long-term side-effect monitoring is hampered by limited reporting channels, varied patient adherence to drug treatments, and a lack of self-reporting of side-effects. This paper recommends methods for automatically monitoring health care forums for mentions of adverse drug side-effects (ADEs) that might otherwise take much longer to identify.

1.1 A note on interdisciplinarity

This research project is built on methods and ideas from multiple disciplines to address a public health problem. As the "Theory" and "Methods" chapters will demonstrate, the fields of linguistics, computer science, and bioinformatics are all heavily drawn upon. Certainly,

the routines developed and used here would not fit within just one of those fields; but, by taking an interdisciplinary approach to the problem of discovering new ADEs, a novel approach can come to fruition.

1.2 Thesis plan

The rest of this thesis is broken down into the following sections:

1. The background chapter. This chapter will provide the underpinnings of tools and routines used in this paper, as well as give a brief overview of previous work in ADE extraction.

2. The theory chapter. This chapter discusses the theory behind the methods used here and how they differ from previous work, specifically describing the roll of small-data analysis combined with big-data-driven computational techniques, as well as how the approach proposed here moves away from the need for a side-effect dictionary.

3. The methods chapter. This chapter describes the actual approaches and routines used to perform ADE extraction and analysis of results.

4. The results and discussion chapter. This chapter gives the results of the proposed methods and briefly interprets them.

5. The conclusions chapter. This chapter provides further interpretation of the results and provides recommendations for future work in ADE extraction.

This thesis will outline the journey from foundational work to the realization of theories, routines, and eventually results that contribute to expanding pharmacovigilance efforts.

2 Background

2.1 Aims

This chapter will provide an overview of important terminology, concepts, and routines that will be referenced in later chapters of this paper. It will also describe current methods for reporting ADEs and summarize previous work in post-drug-approval side-effect monitoring (pharmacovigilance), with an emphasis on previous work in mining side-effects from online text sources.

2.2 Current reporting channels

The Food and Drug Administration (FDA) oversees drug production and usage practices within the U.S. This includes regulations on side-effect reporting both during and after the drug is in the clinical trial stage. The FDA requires drug companies to publish a list of all side-effects identified during clinical trials on labels or inserts that come with the drug. In addition, the FDA's post-trial side-effect monitoring (pharmacovigilance) efforts are largely realized through an online reporting system called MedWatch,[4] with a searchable database of reported side-effects called FEARS (FDA Adverse Event Reporting System). This platform allows both patients and health care professionals to report side-effects on their respective web pages. This system, however, is cumbersome and lacking. Reporting of side-effects is almost entirely completed on a voluntary basis. Furthermore, despite FDA efforts to reduce the burden of completing the reporting process,[5] it still requires several pages of information to be completed. An average user likely will not take or have the time to actually complete the form. Most patients in clinical trials understand that there is a

possibility of discovering adverse drug side-effects during the clinical trial period, and that risk is the highest patient-mentioned danger of clinical trial research participation.[6] But many of these same patients may not know anything about post-trial side-effect reporting or even that an FDA website for that purpose exists. While the MedWatch system is constantly monitored and is certainly better than no system, ADE discovery via MedWatch might take years to identify an ADE and can certainly be supplemented by other pharmacovigilance techniques.

Limitations on clinical trial design

Clinical trials are undoubtedly essential to informing treatments and therapies and discovering side-effects. However, there are several issues with clinical trials that limit their overall effectiveness in catching all of the potential long-term issues that a new treatment, such as a drug, may cause in the general population after being approved. One such issue is the size of the clinical trial. Several factors limit the size of clinical trials. These include funding, time, and low success rates.

Clinical trials are extremely expensive. Even for major companies, the total cost of a single clinical trial that progresses through phase III can be over 30 million dollars.[7] Given this high price, it is understandable that companies want to limit costs wherever they can, and it would not be easy to require or convince these companies to include more test subjects and tests for side-effects.

On top of the monetary cost, there is generally a large time commitment necessary to complete a clinical trial. Each phase can take several months to a few years to complete.[8] Time needed to recruit more individuals for the trial or complete more rounds of the trial

would only add to this already large time burden on drug companies and regulatory agencies. Even after spending all of this time testing drugs, however, there is still a low chance that a drug will actually make it to market. Thirty percent of drugs do not even make it past Phase I, and only 25-30% of drugs make it past Phase III.[10] With these low success rates and necessarily limited trial sizes in the early stages, the depth of side-effect discovery is again limited. Even in later trial stages, drug companies still have minimal incentive to expand the trial beyond what is mandated due to the high cost and low probability of success.

Patient adherence and self-reporting of ADEs to health care professionals

Proper patient adherence and usage is obviously critical to ensuring drugs are used correctly and treat patients in the intended way. On the other hand, ADEs can dissuade patients from taking their medication.[9, 10, 11] Beyond causing an ineffective course of treatment, stopping the drug consumption at the wrong time can cause even worse side-effects or even permanent damage to the body.

A patient may hide some of their ADEs from doctors or other health care professionals as well. This is especially true when patients are receiving treatment for high-stakes, serious diseases like cancer.[12] Health care professionals, therefore, cannot report these patients' side-effects. Online health care forums present an opportunity for providing insight into these ADEs. These forums provide a semi-anonymous platform for sharing struggles with diagnoses and drugs, and one post might encourage others with similar experiences to share their negative side-effects as well. Exploring these posts helps to fill the gap in sharing between patients and health care workers or the FDA.

2.3 Natural language processing

Natural Language Processing (NLP) is a central piece of this research. NLP is a well-established field with early work in computer-based language translation[13] and simple artificial intelligence[14] dating back to the mid 1900s. More specifically to this project, modern NLP methods—most critically the Stanford CoreNLP[15] routine—are widely used for large-scale corpus analysis. Broadly, NLP tools are used in this project to extract indicator keywords, find related words that may be side-effects, clean data, and filter results. Combined with other methods described later, NLP facilitates extraction of ADE phrases and analysis of trends within the extracted data.

Stanford CoreNLP

As mentioned above, the Stanford CoreNLP routine plays a critical role in this analysis. Stanford CoreNLP allows text files to be passed through its Java routine and will output a table that provides useful information about each word such as part of speech, Named Entity Recognition (NER) tags, and dependency relationships to other words. The specific implementation of the Stanford CoreNLP routine is further described in the "Methods" chapter.

Dependency trees

A dependency tree is a linguistic tool that provides information about how words in a sentence relate to each other. Typically, each word's part of speech is labeled, and words are related to others by arrows labeled with the dependency relationship (DEPREL) between those words. This tool is employed here to extract parts of sentences related to indicator

verbs (see figure 2.1 below). This can be accomplished by recursively finding words that are related to the verb, with the DEPREL being automatically determined by Stanford CoreNLP. This process is outlined in the "Methods" chapter.

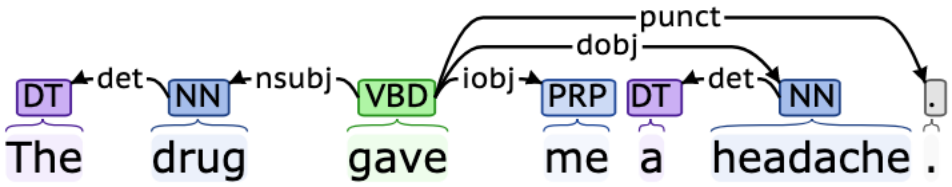

Figure 2.1: Dependency tree example. This image was generated using Stanford's http://corenlp.run/

.

Text classification via SVMs

SVMs are a well-established and ubiquitous method of ML, with early work dating back to over two decades ago.[16] Critically to this research, linear SVMs are effective at classification tasks including text classification.[17] Text classification via SVMs can characterize sample text provided as positive (meeting some pre-defined criterion) or negative (not meeting the criterion) based on training data. Training data for this type of analysis are typically produced by manually tagging positive and negative examples. After learning based on the training data, the SVM will predict, or categorize as positive or negative, a sample or samples you provide. This is especially important in this project for determining whether or not a phrase contains one or multiple side-effects, and (crudely) whether or not they are related to the drug being reviewed.

2.4 Close reading

Close reading refers to the process of manually and carefully analyzing a text or set of texts (corpus). This meticulous parsing of text allows for deep examination of syntax, understanding of syntax and structure, and other details of the text's structure and writing style. When a deep, detailed understanding of one of these elements is beneficial, close reading is valuable and irreplaceable. The main drawback of close reading, and the reason that computer algorithms are necessary for big-data analysis, is that it takes considerable human time and energy. Close reading also tends to focus on tiny pieces of information or specific words rather than examining large-scale trends found in large data sets. Nevertheless, close reading is an essential part of this project. Its specific value here is described in the "Theory" chapter.

2.5 Previous work

There has been significant research performed in the field of side-effect extraction from forums and social media. This section briefly summarizes that research and discusses its limitations. These past research findings also demonstrate the feasibility of many portions of the research presented here, including data gathering, text classification, and eventually extraction of ADEs.

Foundational data gathering

A variety of social media websites and forums have been shown to contain a significant number of ADE mentions, [18, 19] supporting the fundamental source of data in this analysis. Successful text crawling of forums has been demonstrated as well,[20, 18] meaning that data

from these forums can be extracted effectively for the analysis steps used here.

Phrase classification

Other work has focused on classifying phrases as positive or negative for containing an ADE [21, 22, 23]. Classification has been performed based on multiple sources of data, ranging from clinical records to forums.[21] This work in classification provides substantial support for the feasibility of positive and negative classification—including via a SVM—which is the classification method used here, with F-Scores well above 0.8. [22, 23, 24]

Extraction of ADEs

Beyond ADE-related text aggregation and classification, there has also been substantial work in ADE extraction from online data.[22, 25, 26, 27, 28, 29, 30, 31, 32, 33] ADE extraction has been approached in multiple ways in this previous work. Some of the more successful approaches[25, 26] (based on F-Score[18]) often used some form of side-effect dictionary (lexicon) and text classification techniques. Previous research has also demonstrated the value of indicator phrases based on frequency analysis[28] when combined with side-effect dictionaries and ML techniques. While certainly promising based on the results, these methods are hard to generalize to data sets that are not dense with phrases in a dictionary, and require a large amount of training data. The research presented in this paper begins to step away from a dictionary-reliant approach, a process outlined in the next chapter.

2.6 Summary

There are several well-established NLP-related routines that were described in this chapter. Those routines facilitate corpus downloading and cleaning, part of speech and dependency relationship tagging, and text classification. There are also weaknesses in clinical trials and pharmacovigilance efforts by the FDA. Monitoring of health care forum posts can help fill this hole, but current routines are limited by reliance on dictionaries of side-effects, even when using a large data set. The next chapter will present the theory behind a process that will help to move away from dictionary-reliant approaches.

3 Theory

3.1 Aims

This chapter discusses the general theory behind the approach taken in this research project. Key ideas used in the "Methods" chapter will be discussed. The differences in approaches between this and other similar research projects are presented as well. At the end, hypotheses will be formalized as a basis for evaluating results.

3.2 Small data

Big-data analysis is a flourishing topic in both research and popular culture. While often discussed in the context of emerging technology or scientific analysis, the humanities,[34] especially in linguistics, are benefiting from adopting big-data analysis techniques.[35] One such analysis technique is distant reading, in which algorithms scan through massive sets of text, often a pre-defined corpus, with the goal of discovering trends and patterns that humans would not have the time to discover by reading documents on their own. Distant reading is certainly a useful method for parsing large amounts of text, and one that is employed in this research through multiple NLP routines, including Stanford CoreNLP and SVMs (discussed further in the "Methods" chapter). Indeed, the kind of large-corpus analysis needed to extract ADEs in this project would not be feasible without distant reading routines. There are still instances, however, when analysis of small data sets via close reading, even manually, is useful—especially when combined with new visualization and annotation methods.[36] Close reading focuses on specific words and stylistic choices, rather than the large-scale trends. Beyond the enjoyment of a good book or piece of prose, it is argued here that close

reading can provide important fundamental information that, when coupled with distant reading via data analysis algorithms, fosters uniquely effective methods of data parsing and analysis.

3.3 Verbs-first approach

In this project, small data analysis via close reading was performed primarily and most fundamentally during extraction of indicator verbs (described in the "Methods" chapter). Verbs are chosen as the central indicator words here for ADE extraction. This is because verbs facilitate actions and events. It follows that an adverse event caused by a side-effect would be at least crudely indicated by a verb. In the case of this project, a person (typically the one writing the forum post) will experience a sensation, whether it is caused by the drug of interest, another drug, or something else altogether. While determining *what* is fundamentally causing the feeling or sensation is a somewhat tricky task, accomplished here by using a SVM trained on a few hundred data points, verbs that tie together cause and effect are a useful place to begin analysis. Beyond this narrow-scope analysis of cause and effect between a drug and an ADE, analysis of the role of verbs has long been shown to provide insight into structural and stylistic properties as well as improve event classification algorithms.[37]

Once all parts of speech were tagged, verbs were the first target of analysis. Via close reading, a list of frequently-used verbs likely to co-occur with ADEs were extracted (see "Extracting indicator verbs" in the "Methods" chapter). Creating a list of indicator words—the extracted verbs—narrowed the scope of later analysis algorithms and therefore reduced

the time and training data needed to extract the desired information, in this case ADEs mentioned in the forum posts.

In addition to providing a foundation for later steps, initial close-reading results provided an alternative to the current large-dictionary analysis. As described in the "Background" chapter, previous work in extracting ADEs generally required a large dictionary of side-effects and potentially drug names.[1] The analysis described in this paper proposes the beginning of a movement away from large-dictionary-based approaches and towards small-batch, close-reading-based methods of extraction that reduce the need for these dictionaries. As mentioned in the background chapter, past work has shown the value of indicator phrases when combined with other routines[28]. But while the approach used in that previous work relied on a much larger (n=45) set of indicator phrases and a dictionary of side-effects, this work only relies on a small set of verbs (as described in the next chapter).

3.4 Hypotheses

Based on ideas proposed in this section, the following null and alternative hypotheses were formed:

H_0 = Extracting verbs via close reading and using them as the baseline indicator words for side-effects will not be sufficient for extracting ADEs (because it does not find any, gives mostly false positives, etc.)

H_{A_1} = Extracting verbs via close reading and using them as the baseline indicator words for side-effects will allow for extraction of a large number of ADEs, but the effectiveness of

[1]It is worth noting here that the forum used in this research included the name of the drug being reviewed for each post, somewhat reducing the need for a dictionary of drug names. The issue of other drugs causing side-effects appearing in posts still had to be handled.

this extraction approach (measured though recall and F-Score) will be severely inferior to the performance of dictionary-based extraction approaches found in previous literature

H_{A_2} = Extracting verbs via close reading and using them as the baseline indicator words for side-effects will allow for extraction of a large number of ADEs, and the effectiveness of the extraction approach will perform within the range of dictionary-based approaches found in previous literature[22] or better (these methods are competitive with the dictionary-based routines).

3.5 Summary

Previous work in ADE extraction relied on a dictionary of side-effect terms. This paper presents tools for moving away from large-dictionary-reliant analysis, and instead uses only a handful of verbs. The methods presented here are made possible by initial close reading of verbs, and then a movement to distant reading techniques and algorithms that are designed for massive amounts of data. The next chapter presents the specific methods through which this process is realized.

4 Methods

4.1 Aims

This chapter will provide a detailed description of the methodology used for executing the actual data gathering and analysis that follows the theoretical overview mentioned in the previous section. A pipeline will be provided that describes each major step of the analysis, and sub-sections of this chapter will focus on specific approaches to each of these analysis steps. Methods used here will address hypotheses described at the end of the "Theory" chapter, and specific quantitative analysis methods that will be used to test these hypotheses will be outlined.

4.2 Corpus selection and gathering

Forum selection

As mentioned in previous chapters, data from health care forums are at the center of this research. While there are many health care forums available in English, many did not explicitly allow for web scraping or non-commercial use of the data. Others lacked sufficient posts (often less than ten) for many of the drugs being examined. Therefore, the website Drugs.com[38] was selected as the source of data. Specifically, the "User Reviews" section was parsed for reviews of each selected drug.

Drug selection

The first set of prescription drugs examined includes ten of the most commonly prescribed

drugs[2] in the United States[39]. This set of drugs is used because there should be a high number of forum posts about them, and the list of side-effects should be well-established given the high volume of use. This group of drugs includes:

1. Synthroid
2. Crestor
3. Ventolin HFA
4. Nexium
5. Advair Diskus
6. Lantus Solostar
7. Vyvanse
8. Lyrica
9. Spiriva (Handihaler)
10. Januvia

The second set of drugs are newly released prescription drugs approved in 2017 and 2018. To create this list, all drugs approved in 2017 and 2018[40] were put into Drugs.com and drugs that, at the time of data collection, had been reviewed were included. This list includes:

1. Actemra
2. Ajovy
3. Aimovig
4. Biktarvy
5. Dupixent

[2]based on number of monthly prescriptions as of 2015

6. Kevzara
7. Mavyret
8. Ozempic
9. Shingrix
10. Siliq
11. Symproic
12. Trulance

Data downloading

After the text was selected, an algorithm to parse Drugs.com for the "User Reviews" section related to the drugs listed above was created (see appendix B). The parser was written in Python with the BeautifulSoup[41] package. Each review was downloaded into an individual text file containing the review and its duration of use (if available). The name of the drug and the date of the review (if available) were used as the title of each text file.

Data cleaning

After text was downloaded and processed by Stanford CoreNLP to generate a CoNLL table, (this process is described below), rows tagged by the NER algorithm as "PERSON" were manually examined and any rows with identifiers, such as a person's first name, were deleted to protect anonymity. To ensure no errors remained, either from of NER algorithm error or human error, all n-grams[3] extracted at the end were again examined to ensure there were no names or other identifiers included. All final results and figures included in this paper were similarly examined to protect forum-user confidentiality.

[3]An n-gram is a word or sequence of continuous words taken from the health care forum corpus that may contain an ADE. A unigram is one word, e.g. "pain," a bigram is two words, e.g. "stomach pain," and a trigram is three words, e.g. "high blood pressure."

4.3 Initial natural language processing

Once all of the forum posts were downloaded for each drug, the files were separated into groups of older, well established drugs (the ten most prescribed drugs) and newer drugs (the ones approved in 2017 and 2018). Once grouped together, the older analyzed drugs (OADs) and newer analyzed drugs (NADs) were run through the Stanford CoreNLP Java routine[4] via a customized python wrapper.[5] The routine created a merged table in the standard CoNLL format for all forum-post data sets. These tables provide, for each word, the Part of Speech Tag (POSTAG) and Dependency Relationship (DEPREL) to related words in the sentence. In addition to these key pieces of information and the NER tagging described in the previous section, it assigns an ID (index of the word in the sentence, starting at one) to each word in the sentence and a HEAD (value of the current token that points to another word's ID or is zero if it is the head node). Looking at connections between HEADs and IDs, which are described by the DEPRELs, allows for creation of dependency trees that relate words in the sentence to each other. These dependency trees are critical to the analysis described in this paper, and this process will be expanded upon in later sections.

4.4 Extracting indicator verbs

Once the CoNLL table of OADs was generated, all verbs were extracted from the OADs via the POSTAGs in the CoNLL table. The verbs were sorted by frequency and manually examined. Strong candidates for indicating side-effects were then selected from among the most frequent verbs after reading twenty-five reviews to better understand the language used

[4]Using the 10/5/2018 version, see https://stanfordnlp.github.io/CoreNLP/history.html.

[5]See Stanford CoreNLP_GUI_v2.py. This routine was created for a different purpose as well and was a joint effort with colleagues.

on the forum. The list of potential indicators includes the following verbs:

Feel	Lose	Start	Cause	Develop
Experience	Notice	Get	Become	Change
	Give		Have	

Table 4.1: (Lemmanized) indicator verbs

4.5 Extracting candidate phrases via partial dependency trees

To extract relevant words that co-occur in sentences with indicator verbs, a partial dependency tree was created. This "tree," in turn, allowed extraction of n-grams likely containing ADEs. Here it is described as "partial" because it does not involve the dependency relationship tags (except negations) and only involves words with certain parts of speech. Constructing this tree was done in these seven major steps on both the OAD and NAD data sets. These steps were:

1. Select CoNLL table rows containing indicator verbs via LEMMA
2. Get verbs' IDs, HEADs, sentence IDs, and document IDs
3. Get other words within the sentence where the ID matches a verb's HEAD
4. Get all other words where that ID matches back to a HEAD within that sentence (including full CoNLL table row)
5. Filter results by part of speech to filter out many words that would not indicate side-effects, leaving only content words (nouns, verbs, adverbs, and adjectives) and pronouns, prepositions or subordinating conjunctions, and negations (by DEPREL) relating to any of these filtered words by DEPREL
6. Bundle all of these words by sentence (and document) ID and put into unique rows of a new table

7. Add columns that include the drug name, file name, and date based on the document they were extracted from for future analysis (e.g. using SVMs)

It has terrible side effects I am trying get off it	Advair Diskus
I started taking it during summr so my schedule of taking it was rather irregular caused me have insomnia decrease my appetite	Vyvanse

Figure 4.1: Example rows of output after extracting potential ADE rows

As is shown in figure 4.1, each row of the resulting table then contained a column with a phrase that likely included one or more side-effects, broken down into one table with NADs and one with OADs. To further filter the results and remove non-side-effect phrases, a ML technique—specifically a set of support vector machines (SVMs)—was used. The next section describes that process.

4.6 Classification via linear support vector machines

As mentioned in the "Background" chapter, SVMs, specifically linear SVMs, are a widely-used example of machine learning (ML) algorithms that are effective at text classification similar to the classification done in this project. Here, two linear SVMs were employed to accomplish two classification tasks that selected phrases containing ADEs and removed phrases that did not contain ADEs.

Overview of SVM routines

The linear SVMs used for classification were based on scikit-learn[42], a python package with several tools for data mining, data analysis, and ML. More specifically, the LinearSVC, TfidfVectorizer, Pipeline, Model Selection, and Feature Selection routines were imported

and used to clean and convert data, generate and train the SVM, and generate classification predictions on new data (ADE rows described in the previous section). The major SVM-related steps were:

1. Hand-code a subset of previously-extracted candidate OAD phrases as positive (containing one or more ADEs related to the drug being reviewed) or negative (not containing any ADEs related to the drug being reviewed)
2. Train the first SVM (SVM-1) on this data
3. Run predictions on all remaining phrases (n-grams) from the partial dependency tree output for all OAD/NAD phrases
4. Break down the extracted phrases into all combinations of uni-, bi-, or tri-grams (SVM not used here)
5. Hand-code a set of positive (n-gram is an individual ADE) and negative (n-gram is not ADE or contains multiple) examples independently for uni-, bi-, and tri-grams
6. Train second SVM (SVM-2) independently on each n-gram set
7. Based on each of these training results, predict whether phrases in each set (n-gram size) are an ADE or not
8. SVM analysis is finished, filter out n-grams (rows of the output CSV tables) marked as positive for being an ADE

Basic SVM methodology

By way of a general summary, a SVM is a type of ML algorithm that mathematically divides data points into distinct categories (features). SVMs perform this division by calculating a hyperplane that represents this division of data points (the decision boundary). In general, the training data set can be represented by

$$(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n) \tag{4.1}$$

where y_i is the class (side of the desired division) that a data point belongs to, typically represented as -1 or 1.

The hyperplane dividing the points into categories ($y_i = -1$ or $y_i = 1$ in this example) is a the set of points $\vec{x}$ that satisfies

$$\vec{w} \cdot \vec{x} - b = 0 \tag{4.2}$$

where $\vec{w}$ represents the normal (perpendicular) vector to the hyperplane and b as a constant.

Further mathematical techniques used to calculate the hyperplane can be found on the scikit-learn website[43] and in the foundational work on SVMs by Cortes and Vapnik.[44] Details of SVM implementation here are described in the following sections.

Data cleaning

Before each iteration of one of the SVMs, the input data were cleaned and prepared for use through various methods, including separating intermediate result tables by column to extract n-grams and positive/negative coding, making words lower-case, and splitting strings into individual words. Other common cleaning techniques like removing stop words were done in other steps and were therefore not necessary.

TF-IDF vectorizer

Scikit-learn's Term Frequency-Inverse Document Frequency (tf-idf) vectorizer looked at all of the words provided in the corpus and computed a vector of the tf-idf. For each word or set of words, this routine calculated the tf and idf and multiplied them together to get a

value (the tf-idf). Term Frequency is defined as:

$$\frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \tag{4.3}$$

where t represents the term being analyzed, d is the document, and f is the frequency.

These weights were important for deciding which words were most important to focus on in later parts of the SVM calculations. In SVM-1, a sublinear tf routine is used. This results in a metric equal to

$$1 + log(tf_{t,d}) \tag{4.4}$$

being used in place of tf. This replacement is commonplace because it is unlikely that words with much higher term frequencies (e.g. 20 versus 2) should actually be weighted that much more heavily; accordingly, the weight difference is scaled down to a more reasonable level. In SVM-2, the sublinear option is not used because it resulted much less consistent accuracy scores, and the highest accuracy was not as high as the accuracy when using the simple linear method.

Inverse Document Frequency is defined as:

$$\text{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|} \tag{4.5}$$

where N represents the number of documents being analyzed and $|\{d \in D : t \in d\}|$ represents the number of documents where a term (t) appears.[6] Parameters provided to this routine are described later in this chapter (see table 4.2).

[6]True where $\text{tf}(t, d) \neq 0$), otherwise adjusted to $1 + |\{d \in D : t \in d\}|$

Selecting k best

The second major step in the SVMs involved algorithmic selection of features to be preserved and used in the rest of the routines. Based on the output of the tf-idf vectorizer, scikit-learn's SelectKBest method uses a defined algorithm (here f_classif in SVM-1 and chi2 in SVM-2) to choose a user-defined number of features (variables) the algorithm of choice deems most important. Again, details on the implementation are found in table 4.2.

Linear SVC

The third portion of the SVM implementation was performed using Linear Support Vector Classification (SVC), again via scikit-learn. This routine forms the SVM based on a linear kernel and relies on the two previous steps (generating a tf-idf vector and then selecting the best features). The linear classifier works by dividing examples based on their mathematical representations by looking at the shortest distance between positive and negative examples, and places a "line" in the middle with the goal of dividing positive and negative examples without causing any overlap, as outlined in equations 4.1 and 4.2.

Optimizing the SVMs

Based on a trial-and-error approach to optimization, the following set of parameters was provided to the SVMs to facilitate the most accurate text classification.

TfidfVectorizer			SelectKBest			LinearSVC	
Parameter	**SVM-1**	**SVM-2**	**Parameter**	**SVM-1**	**SVM-2**	**Parameter**	**Both**
n-gram range	1-2	1-n*	score_func	f_classif	chi2	C	1.0
sublinear_tf	true	false	k	50	100	penalty	l1
						max iterations	3000
						dual	False

Table 4.2: Parameters of the main sk-learn functions used in the SVMs. More Specific definitions of the parameters can be found on the scikit-learn website.[42] * n is the n-gram size used in that run of SVM-2 (i.e. 1, 2, or 3).

Accuracy scores of SVMs

Accuracy scores for each of the two SVMs are provided in table 4.3 below. It is worth noting that the training set positive/negative example distribution was skewed towards more negative examples in both SVM routines, which makes the accuracy scores somewhat less convincing.

SVM	Samples	Percent Training	Percent Testing	Accuracy Score
SVM-1	550	80%	20%	0.8545
SVM-2	1500 (500 per n)	80%	20%	n=1: 0.9467 n=2: 0.9467 n=3: 0.96

Table 4.3: SVM-1 and SVM-2 accuracies

4.7 Extracting co-occurring side-effects

The final list of identified ADEs was generated by processing the OAD and NAD CSV files outputted by the second SVM. Rows where the SVM tagged the n-gram as an ADE

were kept and rows where the SVM did not tag an ADE were thrown out. These unigram, bigram, and trigram tables were combined by category (OAD and NAD) for further analysis. From there, the tables were broken down by specific drug via the "drug" column (see 4.2). Finally, the ADEs were sorted by frequency for each drug.

stomach pain	Vyvanse	10/2/10	Vyvanse_10-2-2010.txt
stomach growling	Vyvanse	11/23/16	Vyvanse_11-23-2016.txt

Figure 4.2: Example ADE n-gram rows

4.8 Evaluation of results

Three major metrics are used here to analyze the success and effectiveness of the extraction routines. First, the overall precision of SVM classification for each n-gram size is presented. Second, recall metrics are calculated to illustrate overlap between common and severe side-effects found by this project and those identified on drug inserts for OADs. Third, an F-Score measures the overlap of ADEs extracted from an individual forum post with the actual mentions of side-effects in the each of those forum posts.

With F_1 representing the F-Score

$$F_1 = 2\frac{P \times R}{P + R} \tag{4.6}$$

where P is the precision, and R is the recall.

Precision is defined as:

$$P = \frac{TP}{TP + FP} \tag{4.7}$$

where TP represents true positives, and FP represents the false positives.

And recall is defined as:

$$R = \frac{TP}{TP + FN} \tag{4.8}$$

where TP again represents true positives, and FN represents false negatives.

4.9 Summary

Via a close analysis of verbs followed by the implementation of automatic NLP algorithms and SVMs for classification, ADEs can be extracted. This chapter provided specific details about each step in the extraction process and how algorithms were designed and optimized. The ADE extraction process can be summarized well by the following pipeline graphic (4.3).

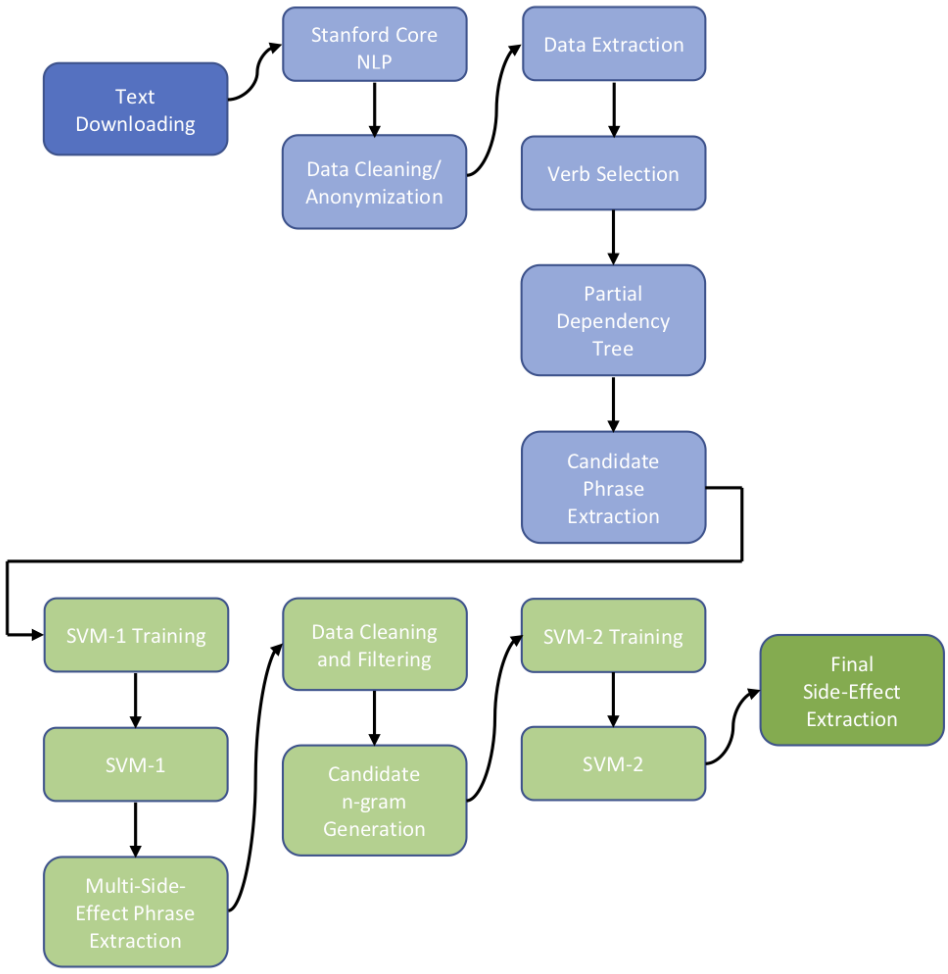

Figure 4.3: Pipeline overview of methods

The next chapter will present and briefly discuss the results from the methods described in this chapter.

5 Results and discussion

5.1 Aims

This chapter describes the results of the analysis methods described above. Information in this chapter includes evaluations of the effectiveness of algorithms via quantitative and qualitative analyses of results, a presentation of newly-discovered ADEs, and a preliminary discussion of how results here fit into the larger body of work in ADE mining. By describing and evaluating results, it is possible to understand benefits and drawbacks of the methods proposed in this paper.

5.2 SVM accuracies

A sample of 45 n-grams was selected from the OAD n-grams tagged as positive matches; more specifically, fifteen samples were chosen for each n (1, 2, or 3) to calculate overall SVM accuracy. True positives represented n-grams that were ADEs, and false positives represented n-grams that did not contain an explicit ADE, n-grams that were unrelated to an ADE, or n-grams with multiple ADEs.[7]

[7]The classifier should only output one ADE at a time (one per n-gram). Parts of longer n-grams should appear in a shorter gram and be accurately classified there (e.g. "stomach pain headache" can be broken down into the unigram "headache" and the bigram "stomach pain."

n-gram Size	Precision
1	0.933
2	0.733
3	0.533
Overall Score	**.733**

Table 5.1: n-gram precision post SVM

The unigram extractions had the highest precision, and precision tapered off as the n-gram size increased (see table 5.1). This suggests that the SVM might be harder to optimize for multi-word phrases, or more training data are needed as the n-gram size increases. Not enough training or test data may explain why the trigram run of SVM-2 produced the highest accuracy but produced the lowest precision here. The sample size for each n here is low, however, so further analysis would be beneficial in understanding why larger n-grams had lower precision scores if the trend holds.

5.3 Overall outputs

Final n-grams that were extracted were mostly related to ADEs as intended. Figure 5.1 below shows the top 100 words in the extraction.

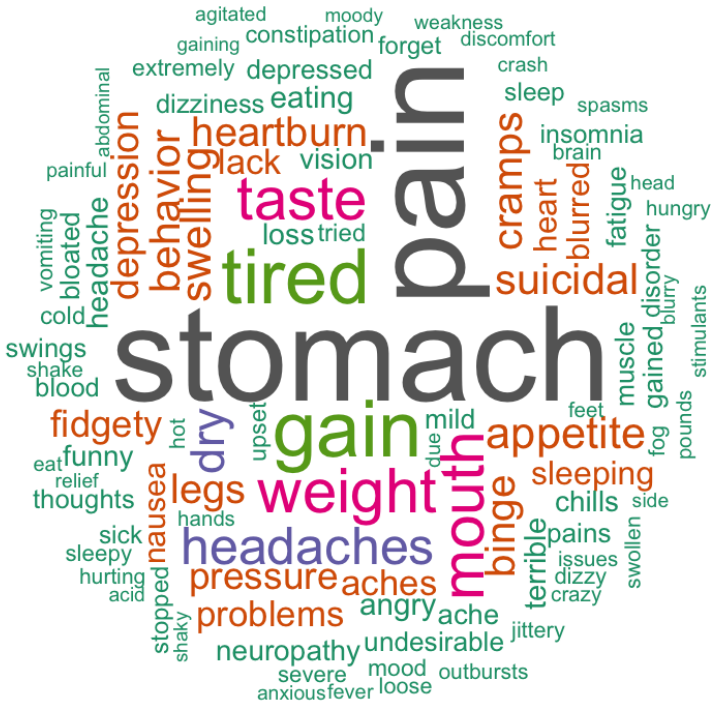

Figure 5.1: Word cloud of extracted ADE n-grams. This contains n-grams of all sizes and for both drugs.

All n-grams that were extracted, sorted by frequency and by drug, can be found in appendix A.

5.4 Comparisons to drug inserts

For each OAD, a recall score was generated to quantify the overlap between ADEs found via forum extraction and the side-effect list on the drug insert.[8] True positives represented ADEs found by the extraction and on the drug insert. False negatives represented ADEs on the side-effect list that were not found by via automatic extraction. The following table includes the recall score for each OAD, as well as the number of samples, or forum posts

[8]The "insert" used here comes from the safety label data on the drug's official website.

downloaded, for that drug and the number of both severe and common ADEs listed on the drug insert.

Drug	**n Samples**	**n Comparisons**	**Recall**
Advair Diskus	113	Common: 7	0.60
		Severe: 12	0.17
Crestor	95	Common: 5	0.57
		Severe: 1	0
Januvia	36	Common: 7	0.33
		Severe: 1	0
Lantus Solostar	21	Common: N/A*	
		Severe: 10	0
Lyrica	789	Common: 8	0.88
		Severe: 3	0.33
Nexium	123	Common: 8	0.63
		Severe: 7	0
Spiriva	77	Common: 13	0.23
		Severe: 7	0.14
Synthroid	149	Common: 18	0.17
		Severe: N/A*	
Ventolin HFA	77	Common: 9	0.22
		Severe: 7	0
Vyvanse	730	Common: 15	0.80
		Severe: 7	0.23

Table 5.2: ADE extraction versus insert recall. *N/A was used when the side-effect list provided on the website did not contain any clearly marked side-effects in that category.

The recall scores here are highly variable, ranging from 0.22 to 0.88. The recall of severe ADEs tended to be much lower, which might be expected because those side-effects are often uncommon (as the drug inserts mention), and sometimes deadly or indicators of deadly conditions, meaning that someone might not have the capacity or time to post about them online on their own.

Among the less severe and more common ADEs, the recall tended to be higher among drugs with a larger number (several hundred) of sample forum posts. This is likely because as the number of people posting about a drug increases, the more likely one of them is to post about their experience with a common ADE.

5.5 Comparing extractions to posts

Thirty posts were selected at random from both the OAD and NAD sets. If the post (text file) being examined did not contain mentions of ADEs and the table of extracted ADEs did not contain any false positives, then it was discarded and another file took its place. True positives represented ADEs found in both the selected forum post and by the extractor (within that same post). False negatives represented ADEs found in the forum text but not by the extractor. False positives represented ADEs found by the extractor but not in the forum post. The F-Score results are in table 5.3 below.

Precision	**Recall**	**F-Score**
0.91	0.51	0.65

Table 5.3: ADE extraction versus original post F-Scores.

The recall and F-Score certainly represent a far from perfect extraction algorithm. Across

the sample of forum posts, there were only four false positives leading to a precision of 0.91, suggesting that the SVMs might have been too strict in classification of positive phrases and n-grams, contributing to a low recall. It might also hint that more indicator verbs would increase discovery of phrases containing ADEs without compromising precision. More positive samples may be necessary in training to improve SVM predictions. While this F-Score appears quite low, it is put into context with other ADE-extraction algorithms in the "Conclusions" chapter, and is reasonably competitive with many of the past dictionary-based ADE extraction results. It is also worth noting that over 85% of the time, the ADE was still identified for that drug via this system, even if it was not in the same file.

5.6 Comparison with FEARS

As mentioned briefly in the "Background" chapter, the FEARS database provides a way to search for ADEs that have been discovered and reported (to the FDA). All extracted NAD ADEs were compared to the FEARS database to look for any newly discovered ADEs not yet indentified in FEARS. The ADEs were manually "verified" as a truly reported ADE related to the drug before being put into this list (by looking for the ADE in the file it was extracted from). The following ADEs for NADs have not been identified in FEARS but were results of the extraction routines:

Drug	Discovered ADEs
Actemra	Heart attack
Aimovig	Cramps
Biktarvy	Bad taste
	Increased Cholesterol
Dupixent	Fever
Mavyret	Hair loss
Ozempic	Stomach cramps
Shingrix	Chills
	Leg weakness
	Vomiting
Trulance	Bloating

Table 5.4: Newly discovered NAD ADEs (The ADEs discovered that were not in the FEARS database)

These discoveries further demonstrate the value of continuous monitoring of forums and an early success of the small-data driven routines proposed by and tested in this paper.

5.7 Summary

By using verbs rather than a large dictionary of side-effects, meaningful ADEs can be extracted from health care forums. It remains possible to identify previously discovered ADEs, as shown by table 5.2, with success increasing as the number of reviews increases. In

addition, there is a high recall score when comparing ADEs identified by the extractor with ADEs actually mentioned in forum posts. The precision is not as strong, but there is still a significant amount of overlap between extracted and mentioned ADEs. New ADEs related to NADs were also identified. The next section expands upon the brief discussion of this section and provides recommendations for further work.

6 Conclusions

6.1 Analysis of findings

Based on the numerical analysis found in the "Results and discussion" chapter, the methods here certainly show promise for facilitating automatic ADE extraction from health care forum posts. Several new ADEs were discovered for NADs here, arguing for the importance of continuous forum monitoring. Though only tested on a small amount of data with actual reviews as the comparison source, the F-score here was competitive with other dictionary-based approaches to ADE extraction found in the literature.[18] Therefore, the null hypothesis outlined in the theory chapter is rejected and H_{A_2} is supported. Using close reading of a small data set at a critical step in the analysis allowed for effective big-data extraction without the need for searching via a side-effect dictionary. This approach is exciting because it is not reliant on existing ADE lists (dictionaries), and therefore does not require continuous updates to such lists. It was also shown that the indicator words extracted from one data set (OADs) could be useful for similar extraction from another data set (the NADs).

In addition, as mentioned in the previous chapter in connection to table 5.2, it appears that as the number of posts run through the extraction algorithms here increases, the precision also increases. This suggests that more data, perhaps collected via continuous web parsing and analysis, would be beneficial. Indeed, severe ADEs appear continuously over a multi-year period, as demonstrated in figure 6.1 below, further making the case for ongoing pharmacovigilance via automatic ADE extraction from forum posts.

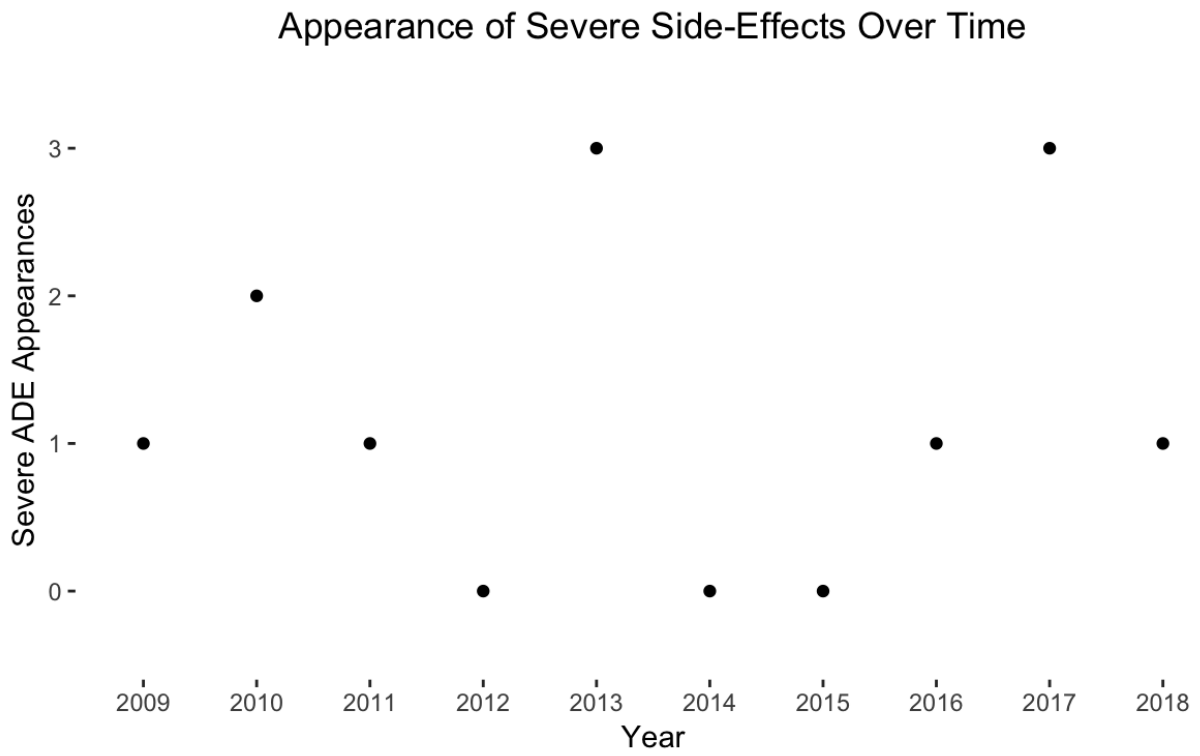

Figure 6.1: Severe extracted ADEs by year. Severe refers to an ADE found in one or more of the OAD drug-inserts that was mentioned as severe.

6.2 Future directions

Even with this early success, there is still much room for improvement. Below is a list of several potential opportunities for such improvement, as well as recommendations for future directions.

- Expanding the set of verbs
- Fine-tuning of SVM classification (comes with more data)
- Deeper analysis of more forum samples (>30 used here) to improve relative F-Score metric, especially as these methods are fine-tuned

- Improving understanding of how side-effects might be described, understated, or overstated on labels and by people on forums
- Adding a spell-checking layer to check for misspellings of drugs, side-effects, or other important words
- Exploring larger n-gram sizes
- Incorporating indicator phrases discovered by others
- Testing non-linear SVMs or other ML techniques like Convolutional or Deep Neural Networks
- Combining these methods with results from previous dictionary methods (until verb-based classification techniques outperform all dictionary-reliant approaches)
- Incorporation of tools for resolution of semantic structure based on verbs like VerbNet
- Use of similar methods to extract user attitudes towards drugs, especially over time
- Extracting ADEs from health care records using similar methods
- Constant monitoring for new ADEs and making the data publicly available (via a website, etc.)

Computing power and breakthroughs in NLP and artificial intelligence have certainly created the potential for new kinds of large-scale data analysis. In combination with current pharmacovigilance infrastructure like FEARS, real-time, automatic health care forum monitoring of any nature can certainly make a meaningful contribution to reduction of ADEs.

ADE extraction from online sources shows great promise, especially if data can be continuously analyzed, shared with regulatory agencies like the FDA, and made public on the internet. Combining close-reading-based, small-data driven techniques with these computational advancements will allow for exciting breakthroughs in health care data monitoring, beyond just extraction of side-effects, and hopefully improve public health interventions to save money and lives.

Appendices

A Extracted ADE tables

CSV tables of extracted ADEs by frequency, separated by drug, can be found at https://jackhester.com/advmine.html.

B Code documents

All code used (R and Python files) can be found at https://jackhester.com/advmine.html.

References

[1] Giacomini Kathleen M., Krauss Ronald M., Roden Dan M., Eichelbaum Michel, Hayden Michael R., Nakamura Yusuke. When good drugs go bad. *Nature.* 2007;446:975–977.

[2] Lazarou Jason, Pomeranz Bruce H., Corey Paul N.. Incidence of Adverse Drug Reactions in Hospitalized Patients: A Meta-analysis of Prospective Studies. *JAMA.* 1998;279:1200.

[3] Health & Ageing Department. Strategic Review of Health and Medical Research. *Australian Government.* 2013:10.

[4] U.S. Department of Health and Human Services. MedWatch: The FDA Safety Information and Adverse Event Reporting Program. U.S. Food & Drug Administration. https://www.fda.gov/safety/medwatch/. Accessed March 1, 2018.

[5] Kessler David A.. Introducing MEDWatch: A New Approach to Reporting Medication and Device Adverse Effects and Product Problems. *JAMA.* 1993;269:2765.

[6] Anderson Annick, Borfitz Deborah, Getz Kenneth. Global Public Attitudes About Clinical Research and Patient Experiences With Clinical Trials. *JAMA Network Open.* 2018;1:5.

[7] Martin Linda, Hutchens Melissa, Hawkins Conrad, Radnov Alaina. How much do clinical trials cost?. *Nature Reviews Drug Discovery.* 2017;16:381–382.

[8] FDA. The Drug Development Process - Step 3: Clinical Research. U S Food and Drug Administration Home Page. https://www.fda.gov/ForPatients/Approvals/Drugs/ucm405622.htm. Accessed December 17, 2018.

[9] Krousel-Wood Marie, Thomas Sheila, Muntner Paul, Morisky Donald. Medication adherence: a key factor in achieving blood pressure control and good clinical outcomes in hypertensive patients:. *Current Opinion in Cardiology.* 2004;19:357–362.

[10] Morisky D. E., Levine D. M., Green L. W., Shapiro S., Russell R. P., Smith C. R.. Five-year blood pressure control and mortality following health education for hypertensive patients. *American Journal of Public Health.* 1983;73:153–162.

[11] Grégoire J, Moisan Jocelyne, Guibert Rémi, et al. Tolerability of antihypertensive drugs in a community-based setting. *Clinical Therapeutics.* 2001;23:715–726.

[12] Pedersen Birgith, Koktved Dorte P., Nielsen Lene L.. Living with side effects from cancer treatment - a challenge to target information. *Scandinavian Journal of Caring Sciences.* 2013;27:715–723.

[13] Hutchins W. John. *The Georgetown-IBM Experiment Demonstrated in January 1954.*;3265:102–114. Springer Berlin Heidelberg 2004.

[14] Bobrow Daniel G.. Natural Language Input for a Computer Problem Solving System.pdf. 1964.

[15] Manning Christopher, Surdeanu Mihai, Bauer John, Finkel Jenny, Bethard Steven, McClosky David. The Stanford CoreNLP Natural Language Processing Toolkit. in

Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations.:55–60Association for Computational Linguistics 2014.

[16] Joachims Thorsten. *Text categorization with Support Vector Machines: Learning with many relevant features.*;1398:137–142. Springer Berlin Heidelberg 1998.

[17] Joachims Thorsten. Transductive Inference for Text Classification using Support Vector Machines. :200–209Morgan Kaufmann 1999.

[18] Sarker Abeed, Ginn Rachel, Nikfarjam Azadeh, et al. Utilizing social media data for pharmacovigilance: A review. *Journal of Biomedical Informatics.* 2015;54:202–212.

[19] Ginn Rachel, Pimpalkhute Pranoti, Nikfarjam Azadeh, Patki Apurv. Mining Twitter for Adverse Drug Reaction Mentions:. :8 2014.

[20] Benton Adrian, Ungar Lyle, Hill Shawndra, et al. Identifying potential adverse effects using the web: A new approach to medical hypothesis generation. *Journal of Biomedical Informatics.* 2011;44:989–996.

[21] Sarker Abeed, Gonzalez Graciela. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *Journal of Biomedical Informatics.* 2015;53:196–207.

[22] Liu Xiao, Chen Hsinchun. A research framework for pharmacovigilance in health social media: Identification and evaluation of patient adverse drug event reports. *Journal of Biomedical Informatics.* 2015;58:268–279.

[23] Yang Ming, Wang X, Kiang Melody. Identification of consumer Adverse Drug Reaction messages on social media. *Proceedings - Pacific Asia Conference on Information Systems, PACIS 2013.* 2013.

[24] Bian Jiang, Topaloglu Umit, Yu Fan. Towards Large-scale Twitter Mining for Drug-related Adverse Events. *SHB'12: proceedings of the 2012 ACM International Workshop on Smart Health and Wellbeing: October 29, 2012, Maui, Hawaii, USA. International Workshop on Smart Health and Wellbeing (2012: Maui, Hawaii).* 2012;2012:25–32.

[25] Nikfarjam A., Sarker A., O'Connor K., Ginn R., Gonzalez G.. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association.* 2015:ocu041.

[26] Liu Xiao, Chen Hsinchun. *AZDrugMiner: An Information Extraction System for Mining Patient-Reported Adverse Drug Events in Online Patient Forums.*;8040:134–150. Springer Berlin Heidelberg 2013.

[27] O'Connor Karen, Pimpalkhute Pranoti, Nikfarjam Azadeh, Ginn Rachel, Smith Karen L., Gonzalez Graciela. Pharmacovigilance on twitter? Mining tweets for adverse drug reactions. *AMIA ... Annual Symposium proceedings. AMIA Symposium.* 2014;2014:924–933.

[28] Sampathkumar Hariprasad, Chen Xue-wen, Luo Bo. Mining Adverse Drug Reactions from online healthcare forums using Hidden Markov Model. *BMC Medical Informatics*

and Decision Making. 2014;14:91.

[29] Leaman Robert, Wojtulewicz Laura, Sullivan Ryan, Skariah Annie, Yang Jian, Gonzalez Graciela. Towards Internet-Age Pharmacovigilance: Extracting Adverse DrugReactions from User Posts to Health-Related Social NetworksRobert. in *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing.*:117-25Association for Computational Linguistics 2010.

[30] Nikfarjam Azadeh, Gonzalez Graciela H.. Pattern mining for extraction of mentions of Adverse Drug Reactions from user comments. *AMIA ... Annual Symposium proceedings. AMIA Symposium.* 2011;2011:1019–1026.

[31] Hadzi-Puric J., Grmusa J.. Automatic Drug Adverse Reaction Discovery from Parenting Websites Using Disproportionality Methods. in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining.*:792–797IEEE 2012.

[32] Yates Andrew, Goharian Nazli. *ADRTrace: Detecting Expected and Unexpected Adverse Drug Reactions from User Reviews on Social Media Sites.*;7814:816–819. Springer Berlin Heidelberg 2013.

[33] Freifeld Clark C., Brownstein John S., Menone Christopher M., et al. Digital drug safety surveillance: monitoring pharmaceutical products in twitter. *Drug Safety.* 2014;37:343–350.

[34] Berry David M.. *Introduction: Understanding the Digital Humanities.*:1–20. Palgrave Macmillan UK 2012.

[35] Kirschenbaum Matthew. The Remaking of Reading: Data Mining and the Digital Humanities. 2007.

[36] Jänicke Stefan, Franzini Greta, Cheema Muhammad Faisal, Scheuermann Gerik. On Close and Distant Reading in Digital Humanities: A Survey and Future Challenges. in *Eurographics Conference on Visualization (EuroVis) - STARs*. (Borgo R., Ganovelli F., Viola I.. , eds.)The Eurographics Association 2015.

[37] Klavans Judith, Kan Min-Yen. The Role of Verbs in Document Analysis. *CoRR.* 1998;cmp-lg/9807002.

[38] Drugs.com. Prescription Drug Information, Interactions, & Side Effects. "Drugs.com". https://www.drugs.com/. Accessed January 10, 2019.

[39] Smith,Michael W. The 10 Most-Prescribed and Top-Selling Medications. "WebMD". https://www.webmd.com/drug-medication/news/20150508/most-prescribed-top-selling-drugs. Accessed December 14,
2019.

[40] CenterWatch. FDA Approved Drugs. CenterWatch. https://www.centerwatch.com/drug-information/fda-approved-drugs/. Accessed January 10, 2019.

[41] Leonard Richardson. Beautiful Soup. Crummy.com. https://www.crummy.com/software/BeautifulSoup/. Accessed March 1, 2018.

[42] Pedregosa F., Varoquaux G., Gramfort A., et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research.* 2011;12:2825–2830.

[43] Scikit-learn. Support Vector Machines. Scikit-learn.

https://scikit-learn.org/stable/modules/svm.html. Accessed April 1, 2019.

[44] Cortes Corinna, Vapnik Vladimir. Support-vector networks. *Machine Learning.* 1995;20:273–297.