Building and inferring knowledge bases using biomedical text mining

by

Jake Lever

B.Eng., University of Edinburgh, 2009

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

The Faculty of Graduate and Postdoctoral Studies

(Bioinformatics)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

September 2018

The following individuals certify that they have read, and recommend to the Faculty of Graduate and Postdoctoral Studies for acceptance, the thesis entitled:

Building and inferring knowledge bases using biomedical text mining

submitted by **Jake Lever** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy** in **Bioinformatics**.

Examining Committee:

Steven Jones, Bioinformatics
Supervisor

Inanc Birol, Bioinformatics
Supervisory Committee Member

Sohrab Shah, Bioinformatics
Supervisory Committee Member

Kendall Ho, Emergency Medicine
University Examiner

Roger Tam, Radiology
University Examiner

Additional Supervisory Committee Member:

Art Cherkasov, Bioinformatics
Supervisory Committee Member

Abstract

Biomedical researchers have the overwhelming task of keeping abreast of the latest research. This is especially true in the field of personalized cancer medicine where knowledge from different areas such as clinical trials, preclinical studies, and basic science research needs to be combined. We propose that automated text mining methods should become a commonplace tool for researchers to help them locate relevant research, assimilate it quickly and collate for hypothesis generation. To move towards this goal, we focus on extracting relations from published abstracts and full-text papers. We first explore the use of co-occurrences in sentences and develop a method for inferring new co-occurrences that can be used for hypothesis generation. We next explore more advanced relation extraction methods by developing a supervised learning method, VERSE, which won part of the BioNLP 2016 Shared Task. Our classical method outperforms a deep learning method showing its applicability to text mining problems with limited training data. We develop it further into the Kindred Python package which integrates with other biomedical text mining resources and is easily applied to other biomedical problems. Finally, we examine the applicability of these methods in personalized cancer research. The specific role of genes in different cancer types as drivers, oncogenes, and tumor suppressors is essential information when interpreting an individual cancer genome. We built CancerMine, a high-quality knowledgebase, using the Kindred classifier and annotations from a team of annotators. This allows for quantifiable comparisons of different cancer types based on the importance of different genes. The clinical relevance of cancer mutations is generally locked in the raw text of literature and was the focus of the CIViCmine project. As a collaboration with the Clinical Interpretation of Variants in Cancer (CIViC) project team, we built methods to prioritise relevant papers for curation. Through this work, we have focussed on different ways to extract structured knowledge from individual sentences in biomedical publications. The methods, guidelines, and results developed will aid biomedical text mining research and the personalized cancer treatment community.

Lay Summary

There are too many publications for a single researcher to read. This is particularly true in cancer research where the knowledge can be spread across many journals. We develop computational methods to automatically read published papers and extract important sentences. We first look at co-occurrences, where two terms appear in the same sentence, and build a system for inferring new ones. We then build a system that, provided with enough examples, can extract the meaning from a sentence. This competed in and won a specific problem in the BioNLP Shared Task 2016 community competition. Finally, we use these methods to extract knowledge relevant for personalized cancer treatment, to understand the role of different genes in cancer, and the relevance of different mutations to clinical decisions. Our methods can be generalized to other problems in biology and our results will be kept up-to-date to remain valuable to cancer researchers and clinicians.

Preface

All the work presented henceforth was conducted at Canada's Michael Smith Genome Sciences Centre, part of the BC Cancer Agency, in the laboratory of Dr. Steven J.M. Jones with the collaboration of the Griffith Lab at Washington University in St Louis. I was personally funded by a Vanier Canada Graduate Scholarship, the MSFHR/CIHR Bioinformatics training program, a UBC four year fellowship and funding from the OpenMinTeD Horizon 2020 project. This work was also supported through Compute Canada infrastructure.

A version of Chapter 2 has been published in the Bioinformatics journal and the citation is below. A licence to reuse the text and figures from this paper has been gained from Oxford University Press through the Copyright Clearance Center.

Lever J, Gakkhar S, Gottlieb M, Rashnavadi T, Lin S, Siu C, Smith M, Jones M, Krzywinski M, Jones SJ. A collaborative filtering based approach to biomedical knowledge discovery. Bioinformatics. 2017 Sep 26.

I created the experimental design, did all the analysis and wrote the full initial draft. Dr. Jones came up with the concept of the project. Early versions of the research were undertaken by Martin Krzywinski, Maia Smith, Mike Gottlieb, Celia Siu, Santina Lin and Tahereh Rashnavadi. All authors contributed edits to the final manuscript.

The contents of Chapter 3 have been published as two separate papers listed below. Both papers were presented at BioNLP workshops and are published as part of the ACL anthology. This anthology is made available through a Creative Commons 4.0 BY (Attribution) license which allows for reuse (https://aclanthology.coli.uni-saarland.de/faq).

Lever J, Jones SJ. VERSE: Event and relation extraction in the BioNLP 2016 Shared Task. In Proceedings of the 4th BioNLP Shared Task Workshop 2016 (pp. 42-49).

Lever J, Jones S. Painless Relation Extraction with Kindred. BioNLP 2017. 2017:176-83.

For both these works, I was the main researcher and developed all code and analysis. These works were written entirely by myself and supervised by Dr. Jones.

A version of Chapter 4 has been published on bioRxiv and will be submitted for publication in a journal. It is available with CC-BY 4.0 International license which allows sharing and adaptation.

Lever, Jake, et al. "CancerMine: A literature-mined resource for drivers, oncogenes and tumor suppressors in cancer." bioRxiv (2018): 364406.

I was the primary researcher for this work. Dr. Martin R. Jones and I came up with the concept and Dr. Steven Jones supervised the work. Dr. Eric Zhao and Jasleen Grewal worked on the annotation of data for this work. I developed the methods, annotated data and lead the writing efforts. All authors made edits to the manuscript.

A version of Chapter 5 will be submitted for publication as:

Lever J, Jones MR, Krysiak K, Danos A, Bonakdar M, Grewal J, Culibrk L, Griffith O, Griffith M, Jones SJM, Text-mining clinically relevant cancer biomarkers for curation into the CIViC database

I was the lead researcher for this work. I developed the concept and experimental design for the work. All authors contributed to the writing of the paper. The work was primarily supervised by Drs Obi Griffith, Malachi Griffith, and Steven J.M. Jones. CIViC is supported by the National Cancer Institute (NCI) of the National Institutes of Health (NIH) under award number U01CA209936 to O.L.G. (with M.G. and E.R.M. as co-investigators). M.G. was supported by the NHGRI under award number R00HG007940. O.L.G. was supported by the NCI under award number K22CA188163. The authors would like to thank Compute Canada for the computational infrastructure used.

The Introduction and Conclusion chapters are original work and have not been published or submitted for publication elsewhere.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Short	Long
AMW	average minimum weight
AUPRC	area under the precision recall curve
BRCA	breast cancer
CGC	Cancer Gene Census
CIHR	Canadian Institutes of Health Research
CIViC	Clinical Interpretation of Variants in Cancer
COAD	colorectal adenocarcinoma
CRF	continuous random fields
CUID	concept unique identifier
DO	Disease Ontology
GBM	glioblastoma multiforme
GFP	green fluorescent protein
HCI	human-computer interaction
HMM	hidden markov model
IE	information extraction
IR	information retrieval
KBC	knowledge base construction
LBD	literature-based discovery
LGG	low grade glioma
LIHC	liver hepatocellular carcinoma
LSA	latent semantic analysis
LSI	latent semantic indexing
LTC	linked term count
LUAD	lung adenocarcinoma
MeSH	Medical Subject Headings
MSFHR	Michael Smith Foundation for Health Research
NCI	National Cancer Institute
NER	named entity recognition

NIH	National Institutes of Health
NLM	National Library of Medicine
NLP	natural language processing
NSCLC	non-small-cell lung carcinoma
OpenIE	open information extraction
PMCOA	PubMed Central Open Access
PMID	PubMed identifier
POG	Personalized OncoGenomics
PPI	protein-protein interaction
PRAD	prostate adenocarcinoma
RBF	radial basis function
ROC	receiver operator characteristic
SeeDev	Seed Development
SQL	structured query language
STAD	stomach adenocarcinoma
SVD	singular value decomposition
SVM	support vector machine
TCGA	The Cancer Genome Atlas
TFIDF	term frequencyâ€“inverse document frequency
UMLS	unified medical language system
VERSE	Vancouver Event and Relation System for Extraction
WGS	whole genome sequencing

Acknowledgements

I would like to thank my Ph.D. supervisor, Dr. Steven J. M. Jones, who provided a wonderful environment to explore text mining and cancer genomics. He encouraged me to pursue countless opportunities to present my work and build collaborations. Through this, I learned a great deal about the practicalities of team science that I hope will benefit my career for years to come. I am very thankful that I chose to do my graduate work with him. Many thanks go to Sharon Ruschkowski and Louise Clarke for making my time in graduate school go so smoothly. I would also like to thank my supervisory committee, Drs. Sohrab Shah, Art Cherkasov, and Inanc Birol, for their support and guidance during this research.

I was very lucky to have the opportunity to conduct research alongside the lead researchers of the Personalized Oncogenomics (POG) program. Sharing an office with Drs. Yaoqing Shen, Erin Pleasance, Martin Jones and Laura Williamson was one of the key factors for my enjoyment of my entire Ph.D. Their friendship, anecdotes, and discussions created a wonderful atmosphere. All the other members of the Jones lab, including Eric Zhao, Jasleen Grewal, Luka Culibrk, Celia Siu and Santina Lin, have been good friends whom I hope to stay connected with as our careers progress. They all contributed invaluable work to this research. I am thankful for all the staff at the Genome Sciences Centre and involved in the Personalized Oncogenomics (POG) project for the incredible work that continues to be done. The other members of my Bioinformatics program cohort, including Shaun Jackman and Sarah Perez, have been great friends who helped me settle easily into the program and life in Canada.

I offer Martin Krzywinski great thanks for our conversations and the fantastic opportunities to contribute to the Points of Significance column and several visualization projects. These were wonderful learning experiences in collaboration and communication. His leadership of the coffee club at the Genome Sciences Centre was also an essential component of my research success. I am also thankful for the many conversations with Dr. Morgan

Bye and Amir Zadeh.

I would also like to thank Drs. Obi Griffith, Malachi Griffith, Kilannin Krysiak, Arpad Danos and other members of the Griffith lab at Washington University at St. Louis for all their support and also their hard work that provided data for this work. I would like to thank Dr. Ben Busby at the National Center of Biotechnology Information (NCBI) for his encouragement in my research and for welcoming me to Bethesda for a short research placement during my Ph.D. as a visiting bioinformatician.

Many thanks go to the various funding agencies that have supported this research. This includes a Vanier Canada Graduate Scholarship, a UBC Four Year Fellowship, a scholarship from the MSFHR/CIHR Bioinformatics training program and funding from the OpenMinTeD Horizons 2020 project. The BC Cancer Foundation supports the POG project which has further enabled this research. This work would not have been possible without the incredible online community of developers and researchers who were willing to answer questions, particularly at StackOverflow and RStudio.

Lastly, I would like to thank my family and my partner Chantal for the unending support that I have received.

Dedication

To my mother Ann, my late father David, my sister Sarah and my partner Chantal.

Chapter 1

Introduction

Who was the last person who fully understood all areas of biology and medicine? As the fields have grown, it has become impossible for one researcher or doctor to keep track of the latest research across such broad fields. This question is a popular discussion amongst mathematicians as there are several arguable candidates for the last mathematician who truly understood all the branches of the field at their time. Famous minds, like Euler or Gauss, are commonly cited. Some of the most remarkable work in recent mathematics, such as Andrew Wile's proof of Fermat's Last Theorem (Wiles, 1995), has required the use of multiple branches of mathematics. Uniting knowledge from diverse areas biology will become essential to solving applied medical problems (Altman, 2018, Council and others (2014)).

Getting the right research to the right researchers is a major bottleneck. The topics in the field go from the micro-scale of protein interactions and genetic modifications to the macro-scale of clinical trials and healthcare systems. An individual researcher needs to know more about different areas of biology in order to plan out an experiment and interpret the results. This problem is complicated by the increasing rate of publications in biomedicine (Lu, 2011). These problems necessitate automated text mining methods to help digest and disseminate research results.

The primary driving forces for text mining development are challenges in research communication which are illustrated by three anecdotes. First, Gregor Mendel's seminal genetics work on pea plants was published in 1866 (Mendel and Tschermak, 1866). It lay dormant, acquiring a small number of citations over the following thirty years until being rediscovered in the early 20th century. This indicates the importance of the venue used for publication and clarity of language. Could a cure for an important disease have already been published but not been recognized by appropriate researchers? The second anecdote describes the fear that a huge problem in one field is equivalent to a solved problem in another field. Temple F. Smith and Michael S. Waterman having published the now famous Smith-

Waterman algorithm for local sequence alignment (Smith and Waterman, 1981) discovered the similar problem of aligning stratigraphic sequences in geology. This discovery came through serendipity as the two researchers walked through a geology department and saw a research poster visualizing the alignment problem (Smith, 2015) and not because either of them was involved in geology research. They were quickly able to publish a paper using a similar algorithm for this problem (Smith and Waterman, 1980). The third anecdote describes a case where valuable information is mentioned as a small section of a paper. As it is not a key result of the paper, it is not mentioned in the abstract and overlooked by most researchers. The discovery and understanding of green fluorescent protein (GFP) is an example of this phenomenon. The bioluminescent properties of a protein are discussed only as a footnote in the paper describing the purification of the aequorin protein from the Aequorea jellyfish (Shimomura et al., 1962). These three anecdotes cannot be isolated incidents. There will be numerous further cases of "undiscovered public knowledge" (a term popularised by Dr. Don Swanson (Swanson, 1986b)) where the solution to a research or clinical question already exists within published literature.

Text mining research and the larger natural language processing (NLP) research area use computers to understand human-written text and provide new ways for humans to interact with digital media. Text mining processes large corpora of text for particular types of knowledge that can both direct users towards relevant knowledge and structure the knowledge for easy assimilation. Researchers should use text mining methods in everyday use to collate relevant knowledge and stay up-to-date. The goal of this work is to understand the problems impeding this goal and solve several of them.

One area in which this need to combine knowledge from the macro to the micro is the area of personalized cancer treatment, also known as precision oncology. This approach aims to use genome sequencing data of individual patient tumors to guide clinical decision making. By identifying the genetic mistakes that are causing the uncontrolled cell growth of a tumor and integrating knowledge from across biology, clinicians will be able to understand the reasons behind a cancer's development and hopefully find weaknesses that can be targeted. The knowledge for this is contained within basic biological research studies of protein function and cell biology, larger sequencing studies and statistical analysis, clinical trial data, clinical guidelines and pharmacological recommendations and many other sources of knowledge. Most of this knowledge is contained within published academic literature which is indexed in PubMed.

This thesis develops and carefully evaluates approaches to applying text mining technology for extracting and inferring biomedical knowledge from published literature. We turn these methods to problems faced in personalized cancer treatment research in order to create valuable knowledge bases that condense tens of thousands of papers for easier survey. This combined work moves us closer to a research world where scientists work with text mining tools in order to keep up-to-date with distilled knowledge relevant to their research.

1.1 Objective

The overall objective of this thesis is to develop generalizable methods for extracting and inferring knowledge directly from published biomedical literature that will provide a lasting benefit to both the text mining and larger bioinformatics community. This work will move us one step closer to a world in which researchers use text mining tools and results in their everyday research. The subgoals of the thesis are (1) to explore methods for identifying relations between biomedical concepts (e.g. drugs, genes and diseases) and (2) to apply these approaches to build knowledge bases relevant to precision cancer medicine.

1.2 Background

The following sections will outline the current status of research into biomedical text mining and the relevant problems faced in personalized cancer medicine.

1.2.1 Biomedical text mining

Text mining is the application of informatics to process text documents to retrieve or extract information (Ananiadou and Mcnaught, 2006). In the biomedical field, this can focus on text from published literature, electronic health records, clinical guidelines and any other text source that contains knowledge about medicine or biology. The field broadly focuses on two main applications, information retrieval (IR) for identifying relevant documents and information extraction (IE) for siphoning relevant knowledge in a structured fashion.

In order to extract and structure knowledge from published literature on a large scale, computers must be able to process the raw text. However, computers are designed to deal with numerical data. Text data does not translate well into a form for easy computation. It is stored as a list of characters, either ASCII, Unicode or another encoding. Computers cannot glean any level of understanding from the raw bytes of a sentence. Various steps need to happen in order to build structure from this raw data. These approaches are common in all natural language processing (NLP) solutions and are not specific to the biomedical domain.

The first challenge is generally to split the series of characters into sentences. In most writing, a period is a good predictor of the end of a sentence and rules can be used to catch exceptions. Exceptions include acronyms and titles (such as U.K. and Dr.). These sentences are then split into tokens, generally individual words, which can be treated as independent elements. These tokens can be further processed to identify the part-of-speech (e.g. noun), remove stems (e.g. -ing) and other lemmatization methods (e.g. plurals -> singular). These further steps depend on the downstream analysis to be performed on the data. Statistical systems have been built that integrate knowledge to combine these steps together such as the Stanford CoreNLP parser (Manning et al., 2014). These parsers can then identify substructures within sentences such as noun or verb phrases. Furthermore, additional structure such as dependency parses can build information about how different tokens within a sentence relate to each other (e.g. a noun may be a subject of a verb).

1.2.2 Information Retrieval

Information retrieval (IR) is the task of finding and prioritizing relevant documents for a particular search task. Researchers use these methods daily by searching for academic papers using tools such as PubMed and Google Scholar. Advances aim to improve relevance for search results. A single researcher has a practical limit of the number of papers that they can read in one year, so it is of paramount concern how they select those papers. IR methods can be used to search other text corpora such as clinical guidelines, but the largest research focus is on academic paper retrieval.

Biomedical IR work has benefitted from the approaches developed for web search. Most methods require a set of keywords as input and then return a prioritized list of papers. Older web search tools, such as Altavista,

used direct keyword matching and simple heuristics based on the frequency (Lawrence and Giles, 2000). Search tools encouraged website developers to add hidden metadata into the header information of a web page. Both these methods placed significant trust in the content developers to provide relevant information. Google's Pagerank method dramatically changed how search results were prioritized (Brin and Page, 1998). By treating the web links as a graph, they could model the "importance" of certain websites by the number of websites linking to it.

In the biomedical domain, similar challenges existed for search. Many journals required (and still require) authors to provide keywords for their paper. This data could be used to help indexing and searching papers but were not associated with a standardized ontology. This created inconsistency. In order to solve this, the National Library of Medicine (NLM) developed the Medical Subject Headings ontology (MeSH). This is used to manually annotate all citations in NLM's PubMed indexing service by highly skilled annotators. With this information, PubMed's search can return highly relevant papers for a provided topic. Advanced search functionality allows control of the journals to search, years, authors and many other factors.

For a long time, their search facility ordered results by reverse chronological order. Recent advances have introduced a relevance ranking method that uses different factors including publication type, year and data on how the search term matches the document (Fiorini et al., 2017). A Pagerank-like approach is more challenging in academic literature as the only links between papers are citations which are not truly analogous to links between web pages. PubMed recently implemented a relevance rank system that combined various data types to improve the relevance of the search results (Fiorini et al., 2018).

A similar IR problem to search is the identification of similar documents. In this case, the input is a current document (either published or free text) and the output is a prioritized list of published works that are similar. This document similarity metric is a feature of PubMed through their "Similar articles" option. One solution to this problem uses ideas in document clustering. The basic concept is to group documents that discuss similar topics. The route to extract the topics discussed in a paper can be quite varied. For biomedical abstracts, the associated MeSH terms provide a rich and high quality manually curated resource to allow for document clustering. Simple overlap metrics based on MeSH terms can provide good quality results for similar document classification (Zhu et al., 2009). The textual content of

the document can be interrogated directly. The simplest method groups documents that share similar words. This data is often very sparse as the English vocabulary is very large and similar ideas can be expressed using very different words. Preprocessing methods that standardize case, turn plurals in singulars and other steps can reduce the sparsity. Term normalization can be used to group different synonyms together that describe the same term. Each document is represented by a numeric vector. This is either counts of associated metadata terms or counts of words within the document. This numeric count data is known as a "bag-of-words". To find similar documents, these vectors are then compared often with Euclidean distance or cosine distance.

A popular document clustering method designed for this problem is Latent Semantic Analysis (LSA) (Deerwester et al., 1990) which treats document clustering as an unsupervised learning problem, specifically as a learning by compression problem. It transforms the text documents in a word frequency matrix where documents are along one axis and each word in the vocabulary is along the other axis. Every occurrence of a word i in a document j increments the value of x_{ij}. Hence most of the matrix will be zero. It uses low-rank singular value decomposition (SVD) to compress the sparse data into a small dense space where similar topics will be represented by similar latent variables. One other way to detect similarity between papers is by looking at the similarity of their citations. Papers that cite similar papers, or at least papers with similar topics themselves, likely have similar topics. However, citation networks are challenging to build due to paper and author ambiguity and duplicates (Carpenter and Thatcher, 2014).

Document classification can be invaluable for problems in information retrieval. It uses the content and potentially metadata of a document to predict the specific topic of the document. Similar to document clustering methods it uses word frequencies within the document represented as sparse count vectors. However, as a supervised method, it requires sample documents that have been annotated with specific classes (e.g. the topic of the paper, or whether the document is of interest to the researcher). A traditional binary classifier then attempts to identify the words that make the most accurate predictions. In the biomedical space, there is particular interest in predicting the MeSH terms for a paper to assist in the laborious task undertaken by the National Library of Medicine to annotate all biomedical abstracts with terms from the MeSH ontology. Given the huge number of existing annotated abstracts as training data, several methods have been developed for this task as part of a regular competition, BioASQ

(Tsatsaronis et al., 2015).

1.2.3 Information Extraction

Information extraction (IE) methods identify structured information from a span of text, an entire document or even a large corpus of documents. This allows text to be transformed into a standardized format that can be easily searched, queried and processed by other algorithms. These methods are valuable in the biomedical field for extracting knowledge from published literature, automating the analysis of electronic medical records and many other applications. There are three main problems that information extraction methods try to solve: coreference resolution, named entity recognition and relation extraction.

Coreference resolution addresses the problem of anaphora. Pronouns and non-specific terms are frequently used to refer back to entities named in previous sentences (e.g. "he was first prescribed the drug in 2007"). Coreference resolution attempts to link these terms to their original citation. This can be challenging as there can be many candidate coreferences for a single pronoun in a sentence. For example, the word "it" in a sentence could refer to any of the previous objects mentioned in a document. A naive approach would simply use the most recent noun but this is often wrong. Context must be used to infer which coreferences are most likely (Soon et al., 2001). Furthermore, by processing all coreference decisions at the same time, more optimal solutions can be found that don't create contradictions where the same person is both the subject and object of an inconsistent action (e.g. "she passed her the newspaper') (Clark and Manning, 2015).

Named entity recognition (NER) identifies mentions of specific entities such as genes and drugs. Basic approaches can use exact string matching with a list of entity names (e.g. synonyms of genes provided by the UMLS metathesaurus (Bodenreider, 2004)). NER methods can make use of context within a sentence to predict tokens that would likely be a certain entity type. For instance, a token that comes before "expression" and is all capitals, e.g. "EGFR expression" is likely a gene. NER methods often make use of approaches based on Hidden Markov Models (HMM) or Continuous Random Fields (CRF). These are finite-state based methods that can assign labels to tokens in a sequence provided a set of training data. Exact string matching can provide very high recall but with lower precision due to high levels of ambiguity for frequently used English words (e.g. "ICE" is a gene name, but

is frequently "ice" in non-gene contexts). HMM/CRF methods will provide better precision as they can take the context into account but requires a good training set for the associated entity type. Entity normalization approaches take a tagged entity in a sentence and connect it back to an ontology using the context and a set of synonyms associated with each ontology item. Successful NER tools include BANNER (Leaman and Gonzalez, 2008) for many entity types, DNorm (Leaman et al., 2013) for diseases and tmChem (Leaman et al., 2015) for chemicals.

Relation extraction predicts whether a relation exists between two or more entities provided with text in which these entities appear. These methods may also try to differentiate the type of relationship between these terms (e.g. whether a drug treats or causes a disease). The most basic approach to identify whether a relationship exists between two entities is the use of co-occurrences. At its most basic, this method states that a relation exists between entities if they ever appear within a span of text. The text length can vary depending on the application, but sentences and abstracts are common. This binary decision will lead to very high recall of relations but also likely a high false positive rate.

There are alternative metrics than the simple binary decision of whether a co-occurrence ever appears. Intuitively two terms that appear together in many sentences are more likely to be part of a relationship. When taken across a large corpus of documents, e.g. all publications in a journal or even all accessible biomedical literature, the frequency of co-occurrences can be very high. However, for a single document, these methods may not be applicable. A threshold can be used to cut off co-occurrences that appear too infrequently. These infrequent co-occurrences may be false positives. However, a small number may be valuable information that are simply not commonly discussed.

Co-occurrences will be affected by the frequency of the individual terms. Frequently mentioned terms, such as "breast cancer", will have higher co-occurrence numbers than rarely discussed terms such as "ghost cell carcinoma". Hence a normalization approach that takes into account the background frequency of individual terms can help identify spurious co-occurrences driven by the fact that one or the other term occurs a lot. "Breast cancer" appears in many papers and so is more likely to cooccur with terms. By taking the frequency of the words "breast cancer" into account, we can reduce the false positives. At the same time, we can put greater importance on the few co-occurrences of terms with "ghost cell

carcinoma". This concept is used in the term-frequency inverse-document-frequency (TF-IDF) approach to normalization. Term frequency is the count of terms and inverse document frequency is the normalizer for the frequency of the term in general.

The power of co-occurrences really comes from aggregated information across a large corpus. For individual documents, more advanced relation extraction methods can be used. These can take for the form of supervised approaches (which require substantial example text data), semi-supervised approaches (which require less example data and is easier to acquire) or unsupervised approaches (which use no prior knowledge).

Supervised learning approaches to relation extraction involve a training set of text with annotated entities and relations. The general goal is to transform the text and annotations into a form amenable to traditional classification methods. A common method is to vectorize the candidate relation within a sentence so that it is represented by a numerical (often sparse and very large) vector that can be fed into a standard binary classifier (e.g. logistic regression or support vector machine). These methods use bag-of-words approaches similar to the document clustering discussed previously. This transforms the sentence into a vector representation of word counts. Bi-grams, tri-grams (or n-grams to generalize) capture neighboring two, three, or more words. They can also transform subsections of the sentence, e.g. the clause that contains the relation, or a window of words around each entity. The entity types can also be represented with one-hot vectors (where the vector is as long as the number of entity types with a value of one at the location corresponding to the entity type and zeroes elsewhere). These methods produce very sparse and large vectors and often $p >> n$, where p is the number of features and n is the number of examples used for training. These vectors can then be processed by classifiers such as logistic regression, support vector machines or random forests.

Support vector machines offer an alternative method that avoiding vectorizing the relations. A support vector machine attempts to find the hyperplane that separates the training examples. However, the power of SVMs really comes down to the "kernel trick" which allows SVMs to be solved by using comparisons between training examples instead of vectorizing them and placing them in N-dimensional space. A kernel is simply a similarity function that takes in two examples and returns a similarity value. Without a complex kernel, an SVM is known as a linear SVM and behaves very similarly to logistic regression. Popular kernels include polynomial functions and

radial basis functions (RBF). These kernels implicitly transform the example data into another space where a separating hyperplane is easier to find. For text mining purposes, support vector machines are valuable for the ability of kernel functions to accept example data which aren't numerical vectors. A string comparison kernel can accept two text strings as input and output a similarity measure based on metrics such as Hamming distance or edit distance. This means that a classifier can be built using a similarity measure and no vectorization is required. Furthermore, support vector machines do not require each input example to be compared with every single training example. The SVM identifies the training examples (known as the support vectors) that can be used to define the separating hyperplane. When applied to test data, comparisons are only needed against these "support vector" examples, which allows for a high-performance classifier.

Dependency parsing provides information about the basic relations between words, such as the subjects and objects of a verb and the modifiers that apply to a noun. When these parsers were developed, relation extraction methods quickly began to make use of the information. Bunescu and Mooney specifically argue that the main information about the relationship is contained within the dependency path which is the shortest path between two entities within the dependency parse tree (Bunescu and Mooney, 2005). Kernels that used this information such as the dependency path kernel allows comparison of the dependency parse instead of the full sentence. These use a simple similarity metric based on the number of shared words, parts of speech, and entity types at each place within the two dependency paths being compared.

Deep learning methods have made great headway into non-biomedical information extraction problems with the main computational linguistics research venues being dominated by deep learning methods. These methods exploit the concept of distributional semantics. This is the idea that individual words can be represented as numerical vectors where similar words will have similar vectors. The bag-of-words approach to word representation does not fit this as each word is represented by a one-hot vector which is as a wide as the vocabulary and only has a single one. Each word is therefore orthogonal to all other words in the vocabulary. These techniques depend on large amounts of annotated data as the model complexity of deep learning is very high and methods are liable to overfit. Due to lack of data, deep learning has had a hard time gaining traction in biomedical text mining research.

Event extraction is a special type of relation extraction, sometimes denoted as complex relation extraction. It extracts events described in a sentence which may involve multiple relations. These relations have other relations as arguments instead of entities. There are three relations in this example sentence: "upregulation of one gene decreases phosphorylation of another protein". The upregulation would be one relation, the phosphorylation would be the second relation, and the decrease would be a compound relation connecting the other two relations. Event extraction has been the focus of several shared tasks such as GENIA (Kim et al., 2003). The standard approach involves breaking the task down into a series of binary relation extractions which can be built up into a full event (Björne and Salakoski, 2015).

When fully annotated training data is not available, there are two possible options. Semi-supervised methods use partially annotated data or so-called silver-annotated data. This silver-annotated data is generated using a procedure known as distant supervision (Mintz et al., 2009). When no annotations exist, existing knowledge bases which contain some relevant relations can be used to automatically annotate sentences. For instance, if erlotinib is known to inhibit EGFR, then all sentences which contain both terms could be annotated with this relation. This will produce a larger number of false positive annotations. But if there are enough "seed facts" in the knowledge base, a well-trained classifier may be able to identify the key patterns that link all the sentences and reduce the false positive rate. A fully unsupervised method based on clustering can also be used to group potential relations that look similar. Percha et al grouped relations based on their dependency path and then used a distant-supervision like approach to tag different relation clusters (Percha et al., 2018).

All of these relation extraction methods will annotate a span of text with the location of the relationship and the entities associated with them. Depending on the application, these annotated documents could then be presented to the user, or the relations could be aggregated to allow easier searching. In order to drive research in relation extraction and other areas of biomedical information extraction, there are regular shared tasks organized by the research community. These are competitions where one group releases an annotated training set for other groups to build machine learning systems for. A held-out test set is then used to evaluate the competing algorithms. These competitions have included the BioNLP Shared Tasks (Kim et al., 2011, Kim et al. (2009)), BioCreative tasks (Hirschman et al., 2005) and many others. They provide a good metric of the latest algorithms in the

field. They are especially valuable as biomedical information extraction is hampered by the small annotation sets (compared to non-biomedical domains). Biomedical annotation often requires expert level knowledge and can be difficult to organize. These events encourage the development of methods that can work with few examples.

The documents for these shared tasks are often based on PubMed abstracts and full-text articles from PubMed Central. These resources, which are often used for text mining, are the easiest to access which is a common limiting factor in biomedical text mining. In contrast, it is very difficult to get access to a large corpus of electronic health records which limits the research opportunities in this area. In biomedicine, abstracts are easily accessible through PubMed and can be downloaded in bulk through the NCBI's FTP service. However full-text articles are often challenging to access. The PubMed Open Access Subset provides full-text articles in XML format for over a million full-text articles. This is, however, a fraction of the publications in PubMed. Other researchers have tried mass downloading of the PDFs of published literature. Publishers often limited this in their terms of use contracts and have been known to limit access to their resources for entire universities to encourage individual researchers to desist from mass downloading (Bohannon, 2016). Even with a large set of PDFs, the conversion to processible text is incredibly challenging. PDF is a format designed for standardized viewing and printing across platforms and is not structured for easy extraction of text. Many tools have been developed to try to make this task easier (Ramakrishnan et al., 2012). But with different journal formats, even simple tasks such as linking paragraphs across pages and removing page numbers are challenging.

1.2.4 Applications of Deep Learning

Deep learning methods have exploded in popularity in recent years and have been broadly applied in many fields including computer vision and speech recognition (LeCun et al., 2015). Deep learning involves multilayer and often complex structured neural networks. The backpropagation method which is used to solve the underlying parameters which controls when these artificial neurons fire has been around for several decades (Rumelhart et al., 1986). However the vast aggregation of data in the last decade has seen their performance eclipse other classification methods. It is for this reason that this thesis will not focus on deep learning methods. For high quality results, a very large dataset of annotated data is required. Co-occurrence

data provides very noisy data that can easily be overfit with very complex models. Furthermore biomedical relation data sets are normally counted in the hundreds, perhaps thousands, of annotations which are several orders of magnitude lower than are needed to fully see the benefit of deep learning. Finally, deep learning is also computationally costly which makes it challenging to create a high-quality knowledge base that can be quickly updated.

1.2.5 Knowledge Bases and Knowledge Graphs

Information extraction methods provide a means to extract relations between different entities. By applying these methods to a well-defined problem and using large biomedical text as the input corpus, a variety of knowledge bases have been constructed. These include the STRING database which use co-occurrence methods to identify likely protein-protein interactions (Szklarczyk et al., 2014). The PubTator resource provides automatically annotated PubMed abstracts which are valuable for advanced searching and further text mining efforts (Wei et al., 2013b). An example of information extraction for a very specific domain is the miRTex database which collates information on microRNA targets (Li et al., 2015).

The relations within knowledge bases are often represented as triples. These triples are two entities and the relation that connects them. The set of triples can, therefore, be viewed as a directed graph where vertices are entities and directed labeled edges are relations. Knowledge bases that contain triples can then be queried using SPARQL (Prud'hommeaux and Seaborne, 2006). This is a database query language based on the structured query language (SQL) format used in normal relational databases. The key improvements of SPARQL are the ease of ability to query multiple databases (known as endpoints) and connect together diverse data sets (assuming they can be linked by appropriate unique identifiers).

A growing area of research is inference on knowledge bases. This can involve asking questions of the knowledge base by traversing the knowledge base (Athenikos and Han, 2010). It can also involve making predictions of additions to the knowledge base, particularly new edges to the knowledge base. Most knowledge inference work has focussed on non-biomedical knowledge graphs such as Freebase (Bollacker et al., 2008). The TransE (Bordes et al., 2013) and RESCAL (Nickel et al., 2012) methods focussed on the problem of knowledge base completion (KBC) where there are known to

be edges missing. By using different latent-based approaches, they are able to prioritize missing edges. Several knowledge graphs have been built for biomedical knowledge either through manual curation or automated methods. The WikiData knowledge graph is the structured data backend for all of Wikipedia (Vrandečić and Krötzsch, 2014). It contains a large amount of biological data that is mostly manually curated (Burgstaller-Muehlbacher et al., 2016) and provides a SPARQL endpoint for querying. Other knowledge graphs include KnowLife (Ernst et al., 2014) and GNPR (Percha et al., 2018) which are extracted from text.

1.2.6 Personalized Cancer Genomics

Cancer is a disease of uncontrolled cell growth caused by genomic abnormalities. These abnormalities include small point mutations, copy number variation, structural rearrangements, and epigenetic changes. These affect regulation of growth signaling, control of apoptosis, angiogenesis and many other factors that together are known as the hallmarks of cancer (Hanahan and Weinberg, 2000). These abnormalities can be caused by exogenous mutagens such as smoking or UV radiation, or endogenous mutagens such as oxidation and deamination. Certain chemotherapies can also be mutagenic as damaging DNA can prove lethal to the fast-dividing tumor cells. With the advances in sequencing technology, genomic interrogation of cancers has become commonplace. These investigations are confounded by the driver/passenger mutation paradigm which states that only a small fraction of genomic abnormalities are actually involved in the development of a cancer. These abnormalities (known as drivers) can inactivate key protective genes, or overactivate other genes that normally required careful regulation. The other abnormalities (known as passengers) do not have an oncogenic effect and have "come along for the ride" (Haber and Settleman, 2007).

The goal of personalized (or precision) medicine is to provide a treatment plan that is tailored to an individual patient. This idea holds great promise in cancer treatment as every patient's cancer is different. No two cancers contain the exact same set of genomic abnormalities. By sequencing an individual tumor, researchers hope to identify which genomic aberrations are driver events to understand which pathways are essential to the growth of a cancer. Using this information, combined with knowledge of pharmacogenomics, individualized treatments can be identified.

The Personalized Oncogenomics (POG) project, based at the BC Cancer

Agency, began in 2008. Through whole genome sequencing (WGS) and transcriptome sequencing (RNAseq), the genome and transcriptome are analyzed. Over time, the costs of sequencing have reduced dramatically (Weymann et al., 2017). However, the cost of informatics and genome interpretation have remained stable. This is mostly due to the laborious and manual steps involved in understanding the relevance of important genomic abnormalities within the sequencing data.

There are limited databases that provide some context on whether a likely mutation is a driver or passenger (Forbes et al., 2014) and how to clinically interpret variants (Tamborero et al., 2018). Much of this data is derived from the genomic survey provided by the Cancer Genome Atlas project (Weinstein et al., 2013). This means that analysts must search the vast biomedical literature to understand the latest research for many genes and variants. This area would benefit greatly from the development of new text mining approaches and resources to collate information on the relevance of genes and variants to different cancer types.

1.3 Chapter Overviews

In Chapter 2, we begin by exploring the power of co-occurrences between biomedical terms within sentences. We propose a method for building knowledge graphs using co-occurrences and inferring new knowledge that will likely appear in future publications. With the recent development of recommendation systems, we were inspired to assess a matrix decomposition method against the leading methods in the field. By building a dataset of biomedical co-occurrences from the PubMed and PubMed Central Open Access datasets, we are able to construct a knowledge graph using publications up to the year 2010. A test set is then constructed using publications after 2010 and different prediction methods are compared against it. A comparison of our matrix decomposition method with the other leading solutions to this knowledge inference problem shows that our approach gives dramatically improved performance and provide a step towards automated hypothesis generation for biologists.

Chapter 3 moves past co-occurrences as the method for extracting knowledge and towards full relation extraction based on a supervised learning approach. As part of the BioNLP 2016 Shared Task, we developed a generalizable relation extraction method that builds features from the sentence containing a candidate relation and uses support vector machines. We build upon the

leading work that has shown the power of vectorized dependency-path-based methods. This tool, known as VERSE, went on to win the Bacteria Biotope subtask, came third in the Seed Development subtask and outperformed deep learning based methods. The chapter includes our further development of generalizable relation extraction tools with the Kindred Python package that integrates with many other biomedical text mining platforms including PubTator (Wei et al., 2013b) and PubAnnotation (Kim and Wang, 2012).

Chapter 4 begins to look at applying the information extraction methods to problems faced in personalized cancer treatment. In order to automate the analysis of individual patient tumors, a knowledge base of known drivers, oncogenes, and tumor suppressors is absolutely essential. In order to understand the purpose of a particular genomic aberration, the role of the associated gene must be known for the cancer. Unfortunately, this has previously required manual searching of literature. In this chapter, we describe the development of the CancerMine resource using a supervised learning approach. We hypothesized that the necessary information for drivers, oncogenes and tumor suppressors would be contained within single sentences and that our previously developed methods could be used to extract this information en masse from published literature. To this end, a team of annotators has curated a set of sentences related to the roles of different genes in cancer. By using the methods developed in Chapter 3, we build a machine learning pipeline that can efficiently process the entire biomedical literature and extract cancer gene roles. This data is kept up-to-date and is available to the precision cancer community for easy searching. This data can be integrated into precision oncology pipelines to flag genomic aberrations that are within relevant genes for that cancer type. The annotated set of sentences is also available to the text mining community as a dataset on which to evaluate future relation extraction methods.

Chapter 5 advances our knowledge of clinically relevant biomarkers in cancer. The Clinical Interpretation of Variants in Cancer (CIViC) database is a community-curated knowledge base for diagnostic, prognostic, predisposing and drug resistance biomarkers in cancer (Griffith et al., 2017). This information is invaluable in automating a precision oncology analysis and providing actionable information to clinicians. In order to identify gaps in the CIViC knowledge and prioritize biomarkers that should be curated, we identify published sentences that likely contain all the relevant information. A team of eight curators worked to annotate sentences to link cancers, genes, drugs, and variants as biomarkers. This complex dataset is used to develop a multi-stage extraction system. We provide further advances with a ternary

relation extraction system to integrate drug information. Through validation by the CIViC curation team, we illustrate the power of this methodology for extracting high-quality complex biological knowledge in bulk. This approach is able to provide a vast dataset of very high quality and can easily be applied to other problems in biology and medicine. Furthermore, the dataset of cancer biomarkers is valuable to all groups curating knowledge in precision medicine and also all analysts that are interrogating the genomes of patient tumors.

Finally, Chapter 6 concludes the thesis and discusses the successes and limitations of the research approaches taken. It explores interesting future directions that could be taken with the generalized and high performing methods developed in this thesis and with the valuable precision oncology datasets extracted from the literature.

Chapter 2

A collaborative filtering-based approach to biomedical knowledge discovery

2.1 Introduction

A scientist relies on knowledge contained in many published articles when developing a new hypothesis. Generating new hypotheses automatically based on extracting knowledge from academic publications is the problem faced by literature-based discovery (LBD) algorithms. These approaches are becoming more important as knowledge is spread out across larger number of publications. Text mining tools, including LBD methods, will likely become an essential tool to biology researchers as they explore new research ideas in their specific domains (Ananiadou and Mcnaught, 2006). Most approaches to LBD predict associations between two biomedical concepts that are not frequently discussed in the literature but are predicted to be strongly associated in the future.

Research in the LBD field was first prompted by Swanson's discussions of undiscovered knowledge and his associations of dietary fish oil and Raynaud's disease (Swanson, 1986a). This early technique proposed the concept of open discovery in which a starting term (A) is selected and novel target terms (C) are predicted that are likely associated with A. Swanson's method proposed using intermediate terms (B) that are associated with A and C. For instance, dietary fish oil is mentioned in articles with blood viscosity and vascular reactivity. These two terms are also mentioned with Raynaud's disease. Swanson proposes that it is reasonable that dietary fish oil and Raynaud's disease may be associated, possibly as a treatment. This

result has been validated experimentally (DiGiacomo et al., 1989). William Hersh provides an excellent overview of the different steps involved in the literature-based knowledge discovery problem (Hersh, 2008).

Various tools have been developed to pursue this idea of predicting associations between previously unlinked biomedical terms. All these methods generate a score for a potential association which allow potential associations to be ranked. Swanson's Arrowsmith tool used co-occurrence of biomedical terms in titles from MEDLINE abstracts to identify known associations (Swanson and Smalheiser, 1997). The system required the user to input a starting term, gave them choices on the appropriate intermediate terms and ranked the predicted target terms based on the number of intermediate terms. Co-occurrences have proven a valuable metric for gauging concept associations and have been used in several systems including CoPub (Frijters et al., 2008) and STRING (Szklarczyk et al., 2016). Many other systems have been developed using this concept with different methods for ranking the predictions and most systems generally use the text from the abstract, not just the title. Notable systems include FACTA+ that uses the probability of two terms appearing together in a publication given the frequency of the individual terms (Tsuruoka et al., 2011). The BITOLA system uses the number of intermediate terms as well as the number of papers that support these intermediate links (Hristovski et al., 2013). The ANNI approach uses a comparison of concept vectors to predict novel associations (Jelier et al., 2008b). These concept vectors, based on the symmetric uncertainty coefficient (William, 2007), give a summary of the known associations of each concept with every other concept. The recent Implicitome project makes use of the same methodology as ANNI and has been integrated into the knowledge.bio project (Hettne et al., 2016; Bruskiewich et al., 2016). These methods largely make use of local knowledge, which we define as knowledge of the intermediate terms that cooccur with the starting term and the target terms.

A thorough evaluation procedure has previously been proposed to evaluate the different scoring methods (Yetisgen-Yildiz and Pratt, 2009). It uses publications before a certain year as the input to each approach and evaluates their scoring of novel associations in newer publications. The authors also propose using precision-recall curves as a metric for success which is supported by analysis of the similar link prediction problem (Lichtnwalter and Chawla, 2012).

Recommendation systems are used in many commercial products such as

Amazon and Netflix to suggest relevant products to a customer given their previous purchasing or viewing history. These systems often rely on collaborative filtering algorithms which use the combined history of many users and products. The success of these approaches are largely down to their use of global knowledge with which they can implicitly learn about types of users or products based on this combined history and not any individual user, product or user-product interactions. The Netflix Prize spurred development of new recommendation algorithms and many of the most successful techniques were based on matrix decomposition (Bennett et al., 2007). We propose that similar techniques should be used for literature-based discovery. Instead of associations between users and products, these techniques could be reformulated to predict associations between biomedical terms. They would therefore be able to use global knowledge about the co-occurrence patterns of all entities and be able to implicitly learn about different types of entities. Latent semantic indexing (LSI), a matrix-based approach for finding term similarity, has previously been examined for recapitulating Swanson's fish oil discovery but was limited by computational cost (Gordon and Dumais, 1998).

The literature-based discovery problem can be thought of as a implicit feedback problem (also known as one-class collaborative filtering (Pan et al., 2008). Implicit feedback problems, such as user purchase history, have only positive data points. Missing data may be negative or real missing data. In LBD, we have known associations between biomedical concepts, as they are discussed in the same publications. However the lack of a co-occurrence between two terms can mean two different things: either this is an association that has not yet been discovered, or the two concepts are definitely not associated.

In this chapter, we present the singular value decomposition (SVD) method as the best method for predicting associations between biomedical concepts. We use a similar approach in creating a gold standard data set to the previous comprehensive comparison of knowledge discovery methods (Yetisgen-Yildiz and Pratt, 2009). We build up a training set of co-occurrences extracted from PubMed abstracts and PubMed Central full-text articles up to the year 2010. We then compare methods using their predictions on novel co-occurrences that appear in literature after the year 2010. We also explore the predictive power of this approach to discover associations that appear in literature at various time-points after 2010. Finally, we delve into the several specific associations to examine the strengths and limitations of our SVD method compared to the commonly cited Arrowsmith method.

2.2 Materials and Methods

In order to evaluate different knowledge discovery systems, we extracted a set of co-occurrence relationships to use as training and test sets. These co-occurrences are between different biomedical concepts extracted from the Unified Medical Language System (UMLS) within the same sentence.

2.2.1 Word List

A list of controlled vocabulary terms with synonyms was generated using the UMLS Metathesaurus (version 2016AB - Active Set). The terms selected were filtered from the Semantic Medline groups Anatomy (ANAT), Chemicals and Drugs (CHEM), Disorders and diseases (DISO), Genetics (GENE) and Physiology (PHYS) (Kilicoglu et al., 2008). The Findings group (T033) was removed due to a large number of vague terms. This generated a list of 1,345,346 terms which was filtered using a set of stop words combined from the NLTK toolkit (Bird, 2006) and the most frequent 5,000 words based on the Corpus of Contemporary American English (Davies, 2009). Notably, only ~26% of the terms were found to appear within the downloaded article and abstract text.

All terms in the UMLS Metathesaurus are associated with multiple synonyms and contain alternative spellings and other wordings for the same term. All synonyms were used and matched to a single term ID in the generated word list. When a word (or multiple words) was found in a sentence which was associated with multiple concepts, the co-occurrences were counted for all possible concepts.

2.2.2 Positive Data

Co-occurrence relationships were extracted from biomedical literature to identify potential associations between biomedical terms. Raw text was extracted from titles and abstracts from MEDLINE citations and the titles, abstracts and full texts from PMC Open Access Subset articles. Many relationships may be mentioned in the full paper but not in the abstract (Van Landeghem et al., 2013). Therefore, full articles where available, as well as abstracts, were used to identify the largest possible number of relationships. In total, 13,153,418 abstracts and 1,503,065 full articles were downloaded from MEDLINE and PubMed Central (downloaded through FTP on 12th

Feb 2017). In order to avoid duplication, articles that appear in PMC were filtered out of the MEDLINE data set.

These texts were filtered to remove HTML tags and Unicode special characters. They were split into sentences using LingPipe v4.1.0 (downloaded from http://alias-i.com/lingpipe) and tokenized using the GENIA part-of-speech tagger v3.0.1 (Tsuruoka et al., 2005). Exact string matching was used to identify entities from the UMLS-based word list. Longer terms were extracted first and removed from the sentence. This meant that a sentence discussing "tumor necrosis factor" would be flagged for "tumor necrosis factor" and not for "tumor necrosis". The tokenization was used to identify word boundaries, such that "non-cancerous tumor" was not flagged as "cancerous tumor". When multiple terms appear in a sentence, all pair-wise co-occurrences were recorded.

2.2.3 Sampling and Negative Data

Ideally to evaluate a scoring method, we would calculate the scores for all possible novel co-occurrences, which are defined as co-occurrences that do not appear in the training set. We would then evaluate the difference in scores for known novel co-occurrences in the test set compared with negative co-occurrences, which are those that do not occur in the test set. It is important to note that while all LBD methods discussed in this chapter use only positive data (co-occurrences that do occur in literature) to calculate scores, our evaluation methodology will require the generation of negative data (co-occurrences that neither appear in training or test data).

The training set, from publications published up to and including the year 2010, contains 101,139,316 unique co-occurrences between 305,077 unique biomedical concepts. The size of the set of co-occurrences that could be predicted as novel is ~46.4 billion. The test set contains 65,680,905 novel co-occurrences observed in publications published after the year 2010 and therefore makes up only 0.14% of possible novel co-occurrences.

It is computationally infeasible to evaluate the full space of possible co-occurrences so instead a large sampling approach is taken. 1,000,000 random co-occurrences are selected from the test set that represent known novel associations (also referenced as positive co-occurrences) and do not overlap with the training set. To match the 1,000,000 positive co-occurrences, the same number of "negative" co-occurrences are randomly generated. These

are co-occurrences that don't appear in the training or test data and are very likely not real associations.

2.2.4 SVD Method

The SVD approach treats the co-occurrence data as a binary adjacency matrix X where X_{ij} is 1 if the terms i and j have appeared in a sentence together and 0 if they have not. The matrix is square, symmetric, generally very sparse and has the dimension of the number of terms in the vocabulary. A complete SVD decomposes it into three matrices such that $X = U\Sigma V^T$ where X is the adjacency matrix, U and V contain the singular vectors and Σ is a diagonal matrix containing the singular values.

We use a truncated form of SVD in which we only use a small number of the singular values in order to create a low-rank approximation of the matrix. In this case, we decompose $X \approx U_k\Sigma_k(V_k)^T$ in which we keep the first k singular values and vectors. This means that each term i has a dense representation as the ith truncated singular vectors in U_k and V_k.

By reducing the dimensionality, this approach is able to summarize the original matrix (Eckart and Young, 1936). We used the Graphlab implementation v2.2 (Low et al., 2014) (built from Dato Powergraph Github repository at https://github.com/dato-code/PowerGraph) which uses the Lanczos algorithm. When the truncated SVD is used to reconstruct the matrix, every possible co-occurrence is given a real-valued score which we designate the SVD score. The SVD method gives co-occurrences that are predicted to not appear in future literature a score close to zero, and those that will appear a score closer to one.

There is only one parameter for the SVD method which is the number of singular values k to use for reconstructing the matrix. In order to choose the value for this, we take a cross-validation approach in which we use a further time-split data set. Publications up to the year 2009 are used to generate a co-occurrence training set. And then 1,000,000 novel co-occurrences are randomly sampled from publications in the year 2010. The same negative data generation and sampling approaches are used and precision-recall curves are generated for each rank parameter. By selecting the parameter that gave the largest area under the precision-recall curve, 132 was chosen as the number of singular values.

The SVD method provides scores with a range of approximately zero to one. By setting a different threshold on these scores in order to select the

Table 2.1: Summary of methods for comparison.

Algorithm	Equation for $score(x, z)$
Average Minimum Weight (AMW)	$\frac{1}{\lvert c_x \cup c_z \rvert} \sum_{y \in c_x \cup c_z} min(\lvert c_x \cup c_y \rvert, \lvert c_y \cup c_z \rvert)$
ANNI	$v_x . v_z$
Arrowsmith (LTC)	$\lvert c_x \cup c_z \rvert$
BITOLA	$\sum_{y \in c_x \cup c_z} \lvert c_x \cup c_y \rvert \times \lvert c_y \cup c_z \rvert$
FACTA+	$1 - \prod_{y \in c_x \cup c_z} 1 - D(x, y)D(y, z)$ $D(i, j) = max(P(i\lvert j), P(j\lvert i))$ $P(i\lvert j) = \lvert c_i \cup c_j \rvert / \lvert s_j \rvert$
Jaccard	$\lvert c_x \cap c_z \rvert / \lvert c_x \cup c_z \rvert$
Preferential Attachment	$\lvert c_x \rvert + \lvert c_z \rvert$
SVD	$(U_k)_x \Sigma_k ((V_k)_z)^T$

set of predictions, a trade-off of precision and recall can be made. With $k = 132$, the associated precision-recall curve is examined to identify the optimal trade-off which is equivalent to maximizing the F1-score. We find the score threshold that gives the largest F1-score is 0.44.

2.2.5 Evaluation

Based on previous literature we selected 8 other knowledge discovery algorithms for benchmarking. These methods are based on the number of co-occurrences of terms and occurrences of individual terms. Table 2.1 gives an overview of the equations implemented for the scoring methods. $score(x, z)$ is the score calculated between term x and z. c_i is the set of terms that cooccur with term i. v_i is the concept profile vector as defined in (Jelier et al., 2008a). FACTA+ requires knowledge of the set of sentences that contain term i which is defined as s_i. The SVD method uses truncated versions of the decomposed matrices U, Σ and V. U_k is the truncated U matrix with only the first k columns kept. $(U_k)_x$ is the ith row of the U_k truncated matrix from the SVD decomposition. The same terminology is used for the Σ and V matrices.

The Arrowsmith algorithm counts the number of intermediate terms also known as the linked term count (LTC). The average minimum weight

(AMW) method calculates the path with minimum support between two concepts. An amalgamation of LTC-AMW, in which LTC is used to rank first and then AMW is used as a secondary ranking criterion, was identified as the top performing methods in a previous comparison of literature-based discovery (Yetisgen-Yildiz and Pratt, 2009). We implement LTC-AMW by simply scaling the LTC score up so that the smallest LTC score is larger than the largest AMW score and then add the AMW score and order accordingly. We also compare two successful methods from the link prediction literature, the Jaccard Index and Preferential Attachment (Liben-Nowell and Kleinberg, 2007). Finally we compare three methods from more recent literature-based discovery methods: ANNI, BITOLA and the FACTA+ reliability measure.

A "time-split" approach was used to create a training and test set. This approach has been used previously for literature-based knowledge discovery (Yetisgen-Yildiz and Pratt, 2009) instead of a traditional cross-validation for two reasons. Each data point is not unique as would normally be the case in a classification problem. By randomly assigning each co-occurrence in the training and test sets, the structure of the implicit knowledge graph for training and test would be dramatically altered. The second reason to use the "time-split" method is that it strongly reflects the intended use of these methods, in order to predict future co-occurrences and the so-called "undiscovered public knowledge".

Precision-recall curves were chosen as the evaluation procedure due to the large class imbalance. Previous analysis has shown that receiver operating characteristic (ROC) curves are not appropriate for problems with large class imbalance (Lichtnwalter and Chawla, 2012). When calculating the precision, the prior known class balance , based on the training set, is taken into account. While our test data of positive and negative sampled co-occurrences shows a 50% class balance, the real training data shows a class balance, b, of approximately 0.14% positive co-occurrences within all possible co-occurrences. This information is used to reweight the precision calculation as below where TP is the count of true positives and FP is the counter of false positives.

$$precision = \frac{b \times TP}{b \times TP + (1 - b) \times FP}$$

Recall is calculated as normal and does not require any correction. The F1-score is calculated using the normal recall and the corrected precision.

Table 2.2: Summary of performance for the initial steps for the ANNI and SVD algorithms.

Method	Run-time (h:m:s)	RAM usage (GB)
ANNI Vector Generation	2:31:06	5.8
SVD (with publications up to 2009)	6:21:10	14.8
SVD (with publications up to 2010)	6:09:27	15.6

Table 2.3: Summary of performance for the different algorithms.

Method	Run-time (h:m:s)	RAM usage (GB)
AMW	0:53:15	43.3
ANNI	9:52:13	347.0
Arrowsmith	0:12:58	14.8
BITOLA	0:51:49	43.3
FACTA+	1:30:01	43.4
Jaccard	0:46:18	14.8
LTC-AMW	0:52:22	43.3
Preferential Attachment	0:06:45	14.8
SVD	0:07:32	7.8

2.3 Results

2.3.1 Methods comparison

The 9 methods were compared on the same data set of 2,000,000 randomly sampled positive and negative co-occurrences. In order to visualize the different scoring methods more intuitively, we show violin plots of the various scores for the positive and negative sets in Figure 2.1. The perfect knowledge discovery algorithm would display two separable distributions for the positive and negative sets. However, none of distribution pairs are easily separable showing that none of the algorithms are capable of completely differentiating positive and negative co-occurrences. The performance metrics for the runs of the algorithms are shown in Tables 2.2 and 2.3.

In order to quantitatively compare the different sets of scores, we used the area under the precision-recall curves (AUPRC) which are shown in Fig-

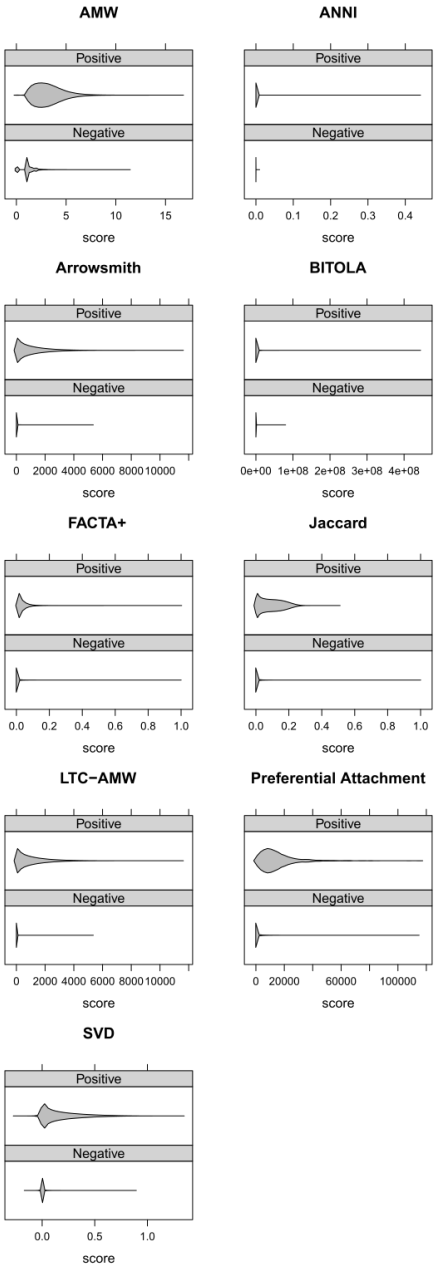


Figure 2.1: Violin plots of the different scores calculated using each method for the positive and negative test co-occurrences shown separately.

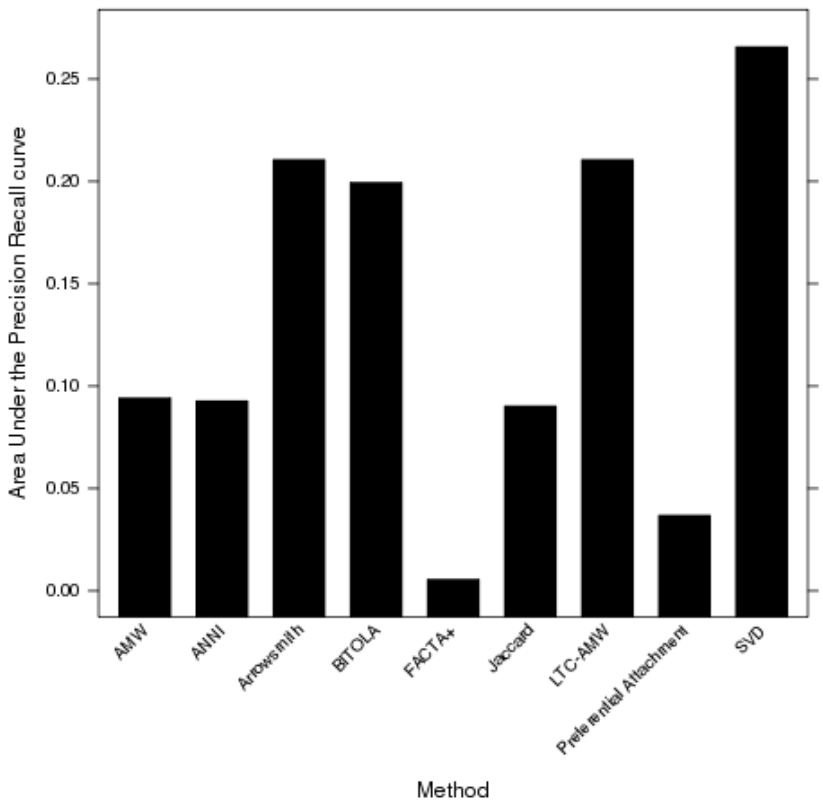


Figure 2.2: The methods evaluated using 1,000,000 co-occurrences extracted from publications after the year 2010, and 1,000,000 co-occurrences randomly generated as negative data.

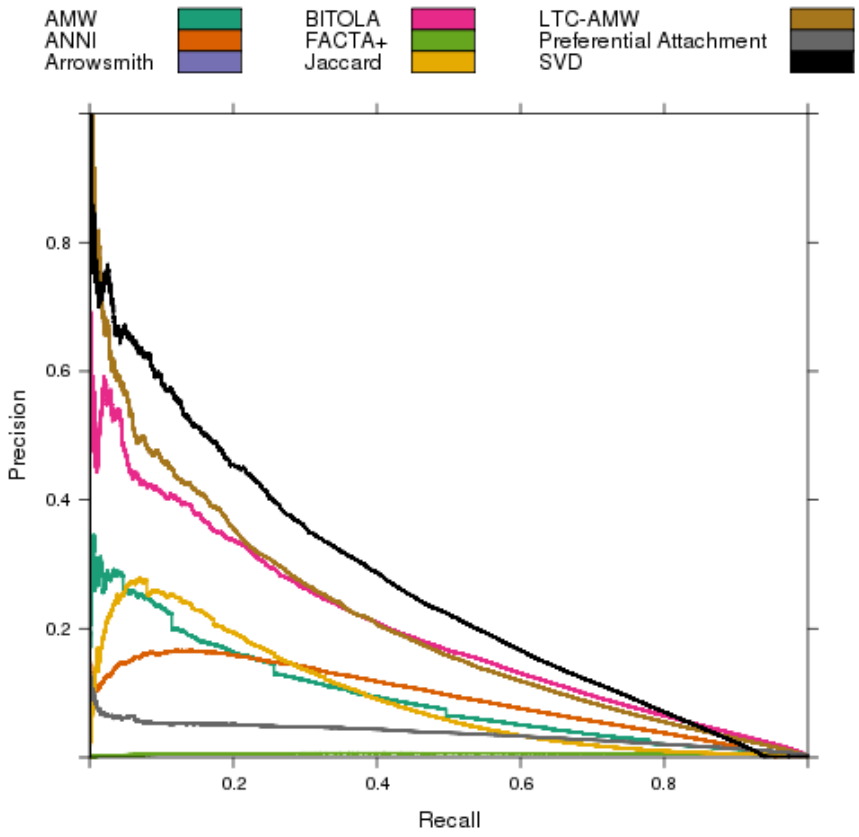


Figure 2.3: The corresponding precision-recall curves for each method shows similar trade-offs for precision and recall for each method.

ure 2.2. Notably, SVD outperforms all the other methods. This suggests that the SVD approach, which is a form of dimensionality reduction, is able to compress the knowledge into a reduced form and generalize the knowledge of the matrix. The associated precision-recall curves, shown in Figure 2.3 highlight that SVD can gain surprisingly high precision if a low recall is acceptable to the user. Arrowsmith gives the second best performance showing that the simple count of intermediate terms gives a strong measure of association between two terms.

While Figure 2.1 suggests that FACTA+ does have different distributions for the positive and negative co-occurrences, the performance shown in Figure 2.2 is surprisingly low. Further analysis showed FACTA+ predicts associations between many extremely rare terms with high probability, a result that disagrees with all other scoring methods. For example, the terms "discorhabdin Y" and "aspernidine A" are predicted to be associated with a probability of 1.0. However both of them only appear in a single sentence each. Given the extreme rarity of these terms, this is a very weak association and likely not helpful. They share a single intermediate term: "alkaloids" that appears in 32,749 sentences, including the single sentences that contain the rare terms. The high probability score is due to the max function used to combine the conditional probabilities $P(i|j)$ and $P(j|i)$ to calculate $D(i, j)$. The conditional probability $P(i|j)$ represents the probability of one term i appearing in a sentence that also contains term j. Given a common term i (e.g. "alkaloids") that occurs in a high proportion of the sentences that a rare term j (e.g. "discorhabdin Y") appears, $P(i|j)$ will be very large and $P(j|i)$ will be extremely small. The max value will always use $P(i|j)$ and these high values skew the results.

The previous comparison analysis (Yetisgen-Yildiz and Pratt, 2009) concluded that the LTC-AMW was the best knowledge discovery method. Our analysis shows the LTC-AMW performs similarly to the Arrowsmith which is equivalent to the linked term count (LTC). This suggests that the improvement of LTC-AMW over AMW previously shown is based entirely on the linked term count and that AMW doesn't contribute at all.

2.3.2 Predictions over time

We also explored predictions for novel co-occurrences that appear in publications at different time points. We again used the data set of co-occurrences from papers up to and including the year 2010. We then found all novel

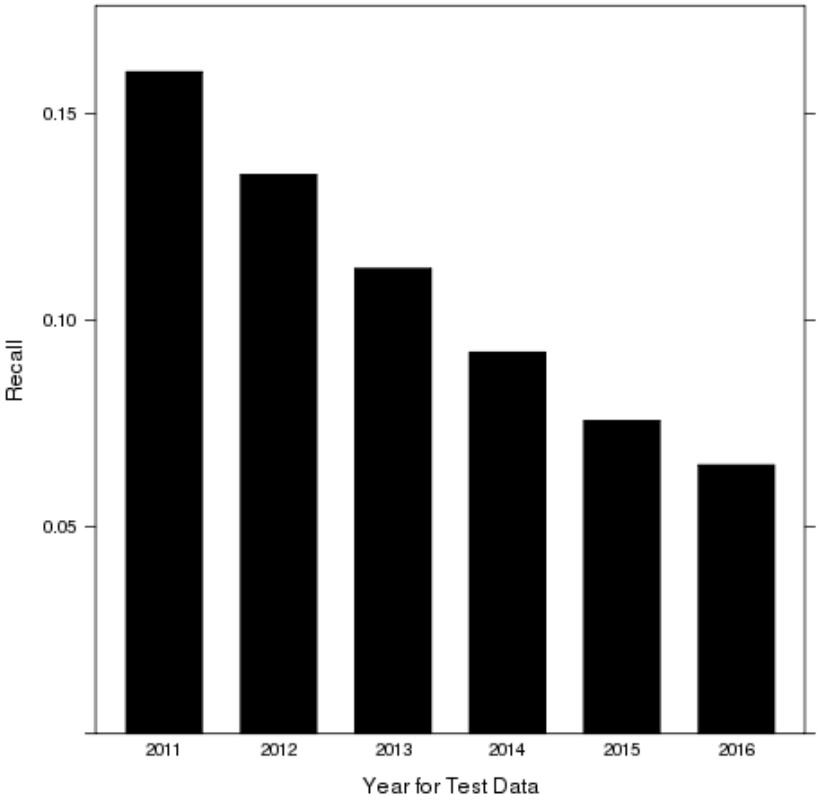


Figure 2.4: Evaluation of SVD predictions on test co-occurrences from publications further into the future using recall as the metric.

discoveries after this period and grouped them by the year in which they first appear. There were on average 10.9 million novel co-occurrences in each year from 2011 to 2016 inclusive. Using the optimal parameters ($k = 132$) for the SVD model, we then calculate the scores using 1 million randomly sampled co-occurrences from each year (for computational reasons). Using the previously selected threshold value of 0.44 on the scores to filter out predictions, we calculate the recall values for each year. These are presented in Figure 2.4.

The model is best able to predict co-occurrences in the year immediately after the data set ends (2011). The recall then decreases each year. This means that novel co-occurrences that appear further in the future are harder to predict. This result makes sense as a large proportion of next year's discoveries will be based closely on existing discoveries. This could be a new drug tested on a similar disease to the current use of the drug or a different member of a gene family being associated with the same disease. However, co-occurrences further into the future are based on more complicated interpretations of the current research or, more likely, new research that has yet to be published.

Importantly, this model should not create too many predictions as to overwhelm a researcher and artificially inflate recall values. The SVD approach makes 12,242,242 co-occurrence predictions with a score above the required threshold. This number of predictions seems reasonable as it is smaller in magnitude to the known number of real novel co-occurrences (65,680,905) in the same time period. One further comment is that a number of the predictions that don't match with a novel discovery in the years up to 2016 will likely appear in future years after 2016.

2.3.3 Comparison of predictions between SVD and Arrowsmith methods

In order to explore the strengths and weaknesses of the SVD approach, we examine four results from the SVD system with comparisons to the output of the Arrowsmith system. The Arrowsmith system is used for comparison as it is the second best performing system. The associated UMLS Concept Unique Identifier (CUID) is noted for each term.

The first case examines the highest scoring prediction from the SVD from our test set. This is an association between "Obstruction" (C0028778) and "Structure of anulus fibrosus of intervertebral disc" (C0223087). SVD gives

this association a score of 1.320. The Arrowsmith method also gives this a high score with 1804 intermediate terms. This prediction turns out to be correct and is found in 7 separate sentences in publications after 2010. One of the papers (Kang et al., 2014) discusses using a block (synonym of "Obstruction" term) to interfere with the "annulus fibrosus" as an experimental model. It is common to block or obstruct parts of the spine to understand developmental biology, hence it is understandable that both SVD and Arrowsmith would make this prediction.

The next case to examine is one in which the SVD method predicts an association which is missed by the Arrowsmith method. Here we find all associations with SVD score above the previously defined threshold of 0.44 and seek the association with lowest Arrowsmith score. This is the association of "Proteins" (C0033684) and "hydantoin racemase" (C0168561). This association has SVD score=0.464 and Arrowsmith score=55. The association is also correct as it is found in a publication during the test period. Hydantoin racemase is an enzyme encoded by a gene in several strains of bacteria. It is unsurprising that there would be discussion of the protein product of this gene and that this association would occur. The SVD method likely implicitly identifies that hydantoin racemase is an enzyme as the pattern of co-occurrences between the enzyme and other terms is similar to other enzymes. Other enzymes are commonly discussed with the word "proteins" as most enzymes are proteins. Arrowsmith likely fails to generate a high score because this is an infrequently discussed enzyme (only appearing in 37 sentences in our corpus and cooccurring with 57 other terms). This suggests that the SVD method may be more successful for infrequently discussed terms.

Next we examine a case where the SVD method failed to predict an association that Arrowsmith found. We look for a case where the Arrowsmith score is above the thresholds defined in Table 2.4 but has the lowest SVD score. This association is between "Surgical Flaps" (C0038925) and "MAP2 gene" (C1417006). Note that "Surgical Flaps" also has the synonym "Flap" and "Flaps". Arrowsmith gives this a high score of 2327, but SVD gives a very low score of -0.175. This association is deemed correct as it appears as a positive association in the test set. However the article in which it appears (Chu et al., 2013) uses "FLAP" to refer to a particular protein and not the expected context of surgical flaps. This shows the limitation of using exact string matching to identify biomedical terms using the UMLS set of synonyms. The question remains why Arrowsmith gives a high score, but the SVD method provides a low score. One likely explanation is that the "Sur-

Table 2.4: Thresholds used for different methods to select prediction set.

Method	Threshold
AMW	5.530
ANNI	2.416e-05
Arrowsmith	2188
BITOLA	4355364
FACTA+	0.029
Jaccard	0.192
LTC-AMW	2188.0
Preferential Attachment	34159
SVD	0.441

gical Flaps" term cooccurs with a large number of terms (15,374) of which only 2,327 (~15%) cooccur with the "MAP2 gene" term. The Arrowsmith method only takes those ~15% into account whereas SVD takes into account the complete co-occurrence pattern when predicting associations. Most of these co-occurrences will be related to "flaps" and "surgical flaps" and not to gene/protein related terms.

Lastly we look at the association with the highest SVD score that was deemed a negative association within our test set, that is one that did not occur in any publications within our corpus. This association is between "Kidney Failure, Acute" (C0022660) and "Thalassemia" (C0039730). The SVD method gave this a score of 0.895 and the Arrowsmith also gave a very high score of 2987. Thalassemia is a group of disorders associated with low haemoglobin production. A publication in 2011 (Quinn et al., 2011) notes that "[l]ittle is known about the effects of thalassaemia on the kidney" and goes on to study the association of thalassemia with renal issues and finding strong links. This suggests that this association is a valid prediction and exemplifies the power of knowledge discovery methods to identify valid links between biomedical terms.

These examples have highlighted several strengths and weaknesses of the SVD and Arrowsmith approaches. Firstly Arrowsmith can be confused by very frequently appearing terms (such as the "Flap" term). It can miss infrequently mentioned terms (such as "Hydantoin racemase"). SVD is able to identify important characteristics of a term, even with infrequent mentions (as was the case for "Hydantoin racemase"). On the other hand, SVD

can also be confused by terms that have a lot of synonyms. If one of the synonyms is a frequently occurring and ambigious term, the SVD method can put too much weight on co-occurrences from this synonym. This limitation may be improved with the development of a named entity recognition (NER) system that can distinguish the context for different UMLS terms. A method built upon the NER systems evaluated in (Funk et al., 2014) would be an interesting direction for a future LBD system.

2.4 Discussion

The success of singular value decomposition over the other current methods for knowledge discovery suggests that the matrix deconstruction approach may be the best avenue for further improvements in knowledge discovery. By compressing the co-occurrence information down to a dense representation of each concept (the row U_i of the U matrix that corresponds to term i), SVD is able to deal with the sparsity inherent in the co-occurrence data. Furthermore it deals with two concepts that aren't frequently discussed together but share the same pattern of co-occurrences with other biomedical concepts. An example would be a drug with generic name and brand names as separate terms in the wordlist (e.g. erlotinib and Tarceva). It would be sensible to merge these entities, however, most knowledge discovery techniques would not be able to do this automatically. Because the two concepts share similar co-occurrence patterns, singular value decomposition will decompose them to similar dense representations and make use of both their co-occurrence patterns to predict new associations. From the recommendation systems perspective, this can be viewed as two customers that watch the same genres of movies but have never watched the exact same movie. The matrix decomposition method is able to identify that these customers share similar tastes and use each others' viewing history to make recommendations.

SVD does, however, have several drawbacks. The first is that it is still computationally expensive. Our SVD runs required ~16GB of memory and about 6 hours per run (on a machine with quad Intel E5-4640 processors). This could be ameliorated through trimming very rare terms, thereby reducing the size of the matrix for decomposition. Furthermore, this will become less of a problem as memory costs decrease. Another issue with singular value decomposition is interpretability so that a user can understand why a prediction is made. Classic methods, such as the Arrowsmith approach,

allows the user to view the intermediate concepts that were used to generate the prediction. As there are no intermediate concepts in the SVD model, it is more challenging to display the rationale for prediction. One approach would be to show the concepts with similar dense representations in order to give context to the user of why these two concepts are predicted to cooccur and presents an interesting future direction for research.

There are many general terms in the UMLS word lists, such as "Local Anesthetic", which may not prove to be useful drug associations. One approach would be to attempt to filter these terms out of the word lists entirely. However, it could be argued that these terms are valuable in understanding the context of other concepts, and in creating their implicit relationships. Hence it would likely be more valuable to filter them out later in the process so that they are not shown as predictions but are used during the singular value decomposition.

The evaluation approach of making predictions using a training set and comparing predictions to a test set (as previously used by (Yetisgen-Yildiz and Pratt, 2009) does have several limitations. The most important for a knowledge discovery algorithm is that many of the predictions deemed as false positives may prove to be true positives as new research is published. This limitation is hard to overcome. Knowledge base completion algorithms make use of a ranking evaluation where the ranking of randomly sampled known positive associations within the full set of predictions is calculated (as used in (Lin et al., 2015). This is used to compare systems and avoids the problem of false positives but is also very challenging to interpret correctly. By using a training/test split approach, the associated metrics of recall and precision give a lower limit to the performance of each system which is easier to interpret. However a testing methodology that avoids the issue of negative data really being positive data (that will appear in future publications) but is also easy to interpret remains an open problem.

Each of the systems generates scores for each association and does not make a binary decision. In order to create a finite set of "predictions", a threshold is chosen for each method and those associations with scores above the threshold are selected. The threshold is chosen in the same manner as for the SVD method. Each method is trained using co-occurrences in publications up to 2009 and evaluated on the co-occurrences that appear for the first time in publications during the year 2010. The threshold that gives the best F1-score using this data split method is selected. Table 2.4 shows the thresholds selected for each method. The predictions shown in Figure 2.5 are based on

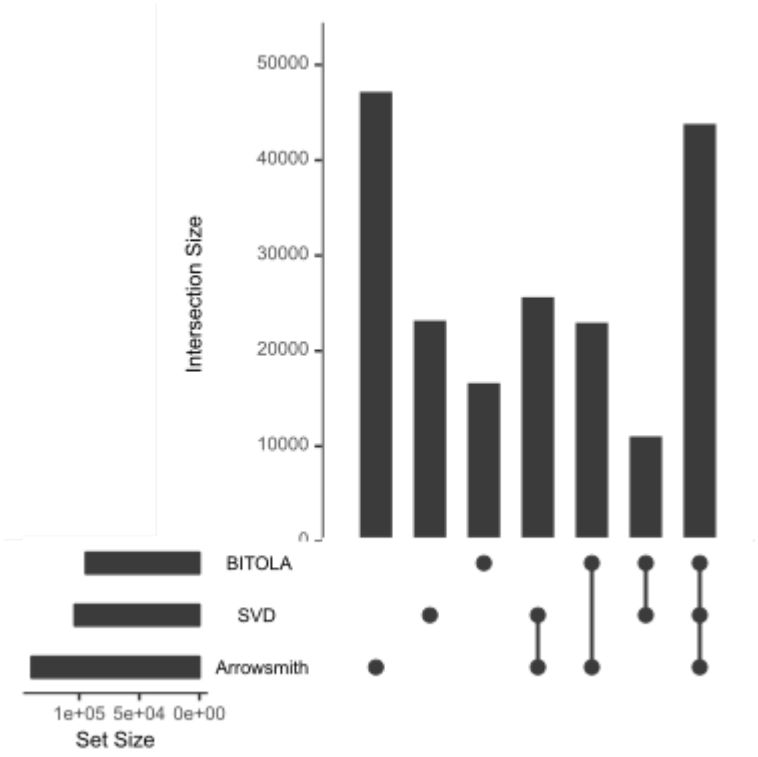


Figure 2.5: An Upset plot showing the overlap in predictions made by the three most successful systems.

the scores generated for the test set of 2,000,000 associations (half of which are positive and half are negative cases). The scores are thresholded and the associations collected for each method.

While the SVD method clearly outperforms the other methods, an obvious question is whether the different systems make similar predictions. Figure 2.5 examines the overlap of top performing systems. LTC-AMW and Arrowsmith give very similar predictions so only Arrowsmith is included. There are a core set of predictions that are shared by each method. However a large number of predictions are made by each system individually. This points towards the development of a meta-method that combines the different predictions of multiple systems and is an interesting direction for future work.

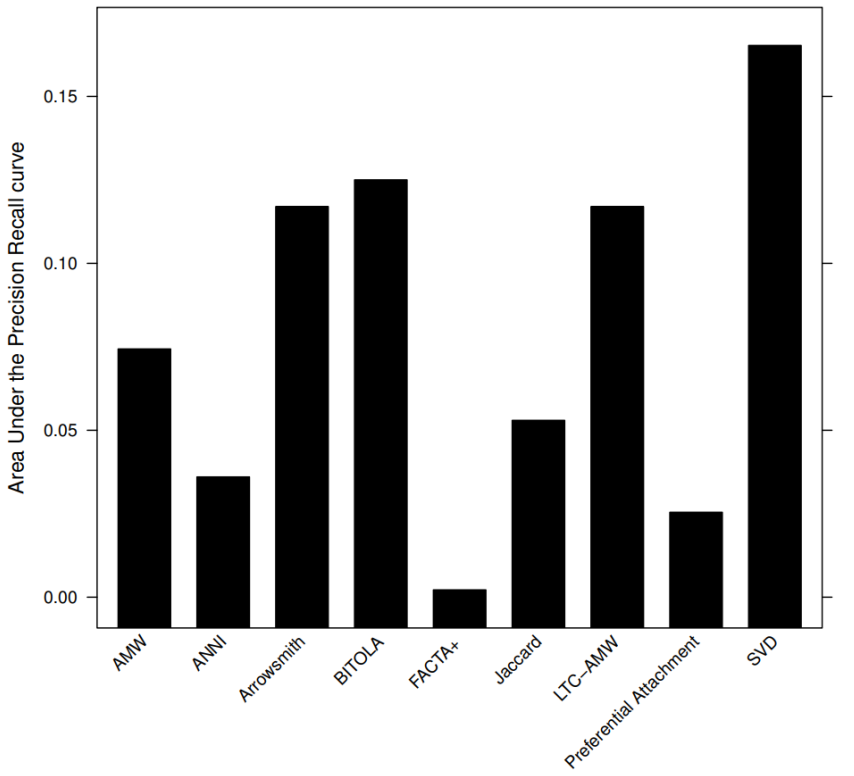


Figure 2.6: The methods evaluated using 1,000,000 abstract-level co-occurrences extracted from publications after the year 2010, and 1,000,000 abstract-level co-occurrences randomly generated as negative data.

It is worthwhile to note that our decision to focus on sentence level co-occurrence, as opposed to abstract level co-occurrence, was based on reducing potential incorrect associations. These happen between terms that cooccur but do not have any real biological relationship. By increasing the amount of text within which a co-occurrence can happen (e.g. to a full abstract), there are likely many more incorrect associations. However to check that this decision didn't bias out results, we reran the entire analysis pipeline using abstract-level co-occurrences. In this case a co-occurrence occurs when two terms appear in the same abstract. The results (shown in Figure 2.6) show a similar pattern to the sentence-level results and that SVD is the best performing system for this type of co-occurrence.

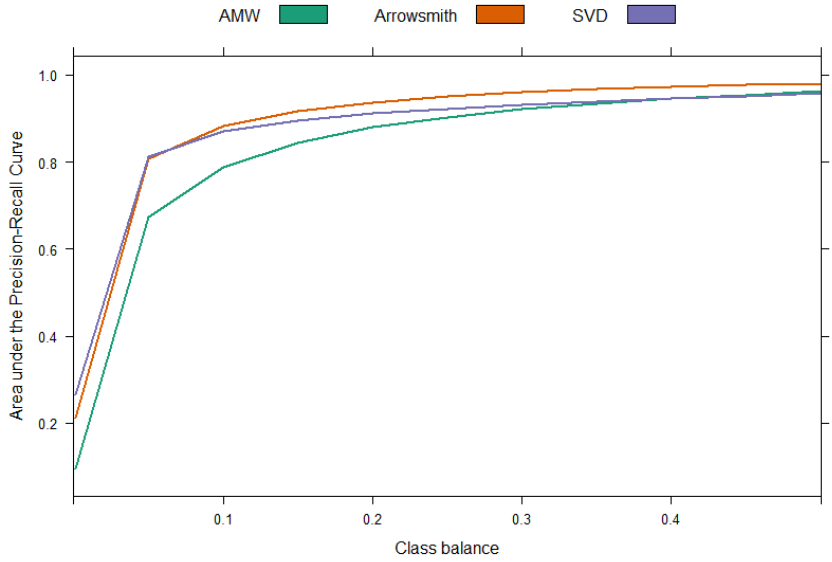


Figure 2.7: The class balance in the dataset can affect the resulting classifier metrics making interpretation of score distributions challenging. The dataset has a class balance of 0.14% which is at the far left. Arrowsmith overtakes SVD at a class balance of ~5% which is an implausibly high class balance of a knowledge discovery dataset.

Finally, we examined the effect that the extreme class imbalance (0.14% positive data) has on the classification metrics. An inspection of the violin plots in Figure 2.1 seems to conflict with the results shown in Figure 2.2. For instance, the AMW results seem to have bulbous positive distribution that has scores clearly larger than the negative distribution. Meanwhile, the

SVD method has an obvious difference between the positive and negative distributions but is not as well defined. But Area under the Precision Recall curve results in Figure 2.2 show that SVD outperforms AMW. We examined the effect that the class balance had on the resulting AUPRC scores in Figure 2.7. This shows that the class balance, which is a property of the dataset, does have an effect on the AUPRC score. This means that visual comparison of score distributions (as in Figure 2.1) is much more challenging. With a very low class balance, more emphasis is put on co-occurrences with high scores. Any false positives with high scores will quickly drop the precision, with a knock-on effect on the AUPRC. This drop increases with larger class imbalance. The ~5% increase in class balance that would be needed to cause Arrowsmith to be the better performing system is very unrealistic for a knowledge discovery problem. Nevertheless, this is an important illustration that the class balance plays an important role in the classification metrics and also in interpreting the score distributions.

2.5 Conclusions

Our study has shown that the singular value decomposition technique provides the best scoring method for predicting future co-occurrences when compared to the leading methods in the knowledge discovery problem. The method is best able to predict co-occurrences that occur in publications in the near future and slowly reduces in predictive power for the far future. We hope this analysis will benefit the knowledge discovery research community in developing tools that will be beneficial for molecular biology researchers.

Chapter 3

Relation extraction with VERSE and Kindred

3.1 Introduction

Extracting knowledge from biomedical literature is a huge challenge in the natural language parsing field and has many applications including knowledge base construction and question-answering systems. In this chapter, we describe our competition winning event extraction system (VERSE) and its followup highly interoperable relation extraction Python package (Kindred).

Event extraction systems focus on this problem by identifying specific events and relations discussed in raw text. Events are described using three key concepts, entities, relations and modifications. Entities are spans of text that describe a specific concept (e.g. a gene). Relations describe a specific association between two (or potentially more) entities. Together entities and relations describe an event or set of events (such as complex gene regulation). Modifications are changes made to events such as speculation.

The BioNLP Shared Tasks have encouraged research into new techniques for a variety of important NLP challenges. Occurring in 2009, 2011 and 2013, the competitions were split into several subtasks (Kim et al., 2009, 2011; Nédellec et al., 2013). These subtasks provided annotated texts (commonly abstracts from PubMed) of entities, relations and events in a particular biomedical domain. Research groups were then challenged to generate new tools to better predict new relations and events in test data.

The BioNLP 2016 Shared Task contains three separate parts, the Bacteria Biotope subtask (BB3), the Seed Development subtask (SeeDev) and the Genia Event subtask (GE4). The BB3 and SeeDev subtasks have separate parts that specialise in entity recognition and relation extraction. The GE4 subtask focuses on full event extraction of NFkB related gene events.

Previous systems for relation and event extraction have taken two main approaches: rule-based and feature-based. Rule-based methods learn specific patterns that fit different events, for instance, the word "expression" following a gene name generally implies an expression event for that gene. The pattern-based tool BioSem (Bui et al., 2013) in particular performed well in the Genia Event subtask of the BioNLP'13 Shared Task. Feature-based approaches translate the textual content into feature vectors that can be analysed with a traditional classification algorithm. Support vector machines (SVMs) have been very popular with successful relation extraction tools such as TEES (Björne and Salakoski, 2013).

3.1.1 VERSE

We will first present the Vancouver Event and Relation System for Extraction (VERSE) for the BB3 event subtask, the SeeDev binary subtask and the Genia Event subtask. Utilising a feature-based approach, VERSE builds on the ideas of the TEES system. It offers control over the exact semantic features to use for classification, allows feature selection to reduce the size of feature vectors and uses a stochastic optimisation strategy with k-fold cross-validation to identify the best parameters. We examine the competitive results for the various subtasks and also analyse VERSE's capability to predict relations across sentence boundaries.

The VERSE method came first in the BB3 event subtask and third in the SeeDev binary subtask in the BioNLP Shared Task 2016. An analysis of the two systems that outperformed VERSE in the SeeDev subtask points to interesting directions for further development. The SeeDev subtask differs greatly from the BB3 subtask as there are 24 relation types compared to only 1 in BB3 and the training set size for each relation is drastically smaller. The LitWay approach, which came first, uses a hybrid approach of rule-based and vector-based (Li et al., 2016). For "simpler" relations, defined using a custom list, a rule-based approach uses a predefined set of patterns. The UniMelb approach created individual classifiers for each relation type and was able to predict multiple relations for a candidate relation (Panyam et al., 2016). This approach of treating relation types differently suggests that there may be large differences in how a relation should be treated in terms of the linguistic cues used to identify it and the best algorithm approach to identify it.

3.1.2 Kindred

There are several shortcomings in the approaches to the BioNLP Shared Tasks, the greatest of all is the poor number of participants that provide code. It is also clear that the advantages of some of the most successful tools are tailored specifically to these datasets and may not be able to generalize easily to other relation extraction tasks. Some tools that do share code such as TEES and VERSE have a large number of dependencies, though TEES ameliorates this problem with an excellent installing tool that manages dependencies. These tools can also be computationally costly, with both TEES and VERSE taking a parameter optimization strategy that requires a cluster for reasonable performance.

The biomedical text mining community is endeavoring to improve consistency and ease-of-use for text mining tools. In 2012, the Biocreative BioC Interoperability Initiative (Comeau et al., 2014) encouraged researchers to develop biomedical text mining tools around the BioC file format (Comeau et al., 2013). More recently, one of the Biocreative BeCalm tasks focuses on "technical interoperability and performance of annotation servers" for a named entity recognition systems. This initiative encourages an ecosystem of tools and datasets that will make text mining a more common tool in biology research. PubAnnotation (Kim and Wang, 2012), which is part of this approach, is a public resource for sharing annotated biomedical texts. The hope of this resource is to provide data to improve biomedical text mining tools and as a launching point for future shared tasks. The PubTator tool (Wei et al., 2013b) provides PubMed abstracts with various biomedical entities annotated using several named entity recognition tools including tmVar (Wei et al., 2013a) and DNorm (Leaman et al., 2013).

In order to overcome some of the challenges in the relation extraction community in terms of ease-of-use and integration, we present Kindred which is a successor to VERSE. Kindred is an easy-to-install Python package for relation extraction using a vector-based approach. It abstracts away much of the underlying algorithms in order to allow a user to easily start extracting biomedical knowledge from sentences. However, the user can easily use individual components of Kindred in conjunction with other parsers or machine learning algorithms. It integrates seamlessly with PubAnnotation and PubTator to allow easy access to training data and text to be applied to. Furthermore, we show that it performs very well on the BioNLP Shared Task 2016 relation subtasks.

3.2 VERSE Methods

The VERSE system competed in the BioNLP Shared Task 2016 and the methods are outlined here.

3.2.1 Pipeline

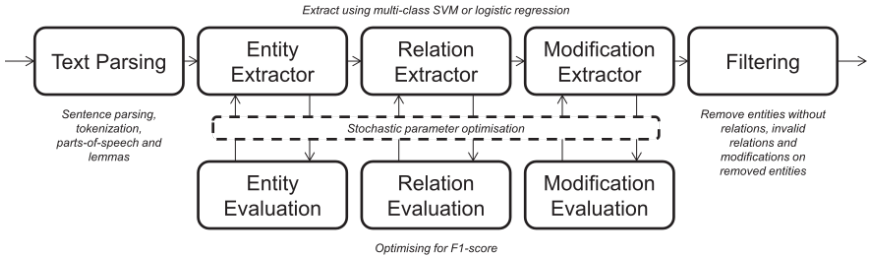


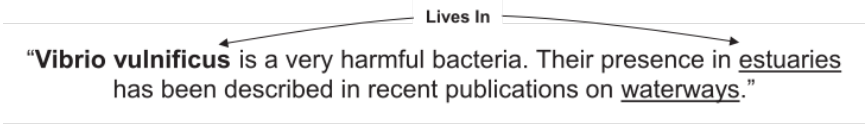
Figure 3.1: Overview of VERSE pipeline

VERSE breaks event extraction into five steps outlined in the pipeline shown in Figure 3.1. Firstly the input data is passed through a text processing tool that splits and tags text and associates the parsed results with the provided annotations. This parsed data is then passed through three separate classifications steps for entities, relations and modifications. Finally, the results are filtered to make sure that all relations and modifications fit the expected types for the given task.

3.2.2 Text processing

VERSE can accept input in the standard BioNLP-ST format or the PubAnnotation JSON format (Kim and Wang, 2012). The annotations describe entities in the text as spans of text and relations and modifications of these entities.

The input files for the shared subtasks are initially processed using the Stanford CoreNLP toolset. The texts are split into sentences and tokenized. Parts-of-speech and lemmas are identified and a dependency parse is generated for each sentence. CoreNLP also returns the exact positions of each token. Using this data, an interval tree is created to identify intersections of text with entities described in the associated annotation. The specific sentence and locations of each associated word are then stored for each entity.



Lives_In	Argument 1	Argument 2
No	vibrio vulnificus	waterways
Yes	vibrio vulnificus	estuaries
No	waterways	vibrio vulnificus
No	waterways	waterways
No	estuaries	vibrio vulnificus
No	estuaries	estuaries

(a) Ignoring argument types

Lives_In	Argument 1 (Bacteria)	Argument 2 (Habitat)
No	vibrio vulnificus	waterways
Yes	vibrio vulnificus	estuaries

(b) Filtering using argument types

Figure 3.2: Relation candidate generation for the example text which contains a single Lives_In relation (between bacteria and habitat). The bacteria entity is shown in bold and the habitat entities are underlined. Relation example generation creates pairs of entities that will be vectorised for classification. (a) shows all pairs matching without filtering for specific entity types (b) shows filtering for entity types of bacteria and habitat for a potential Lives_In relation

Relations and modifications described in the associated annotations are also loaded, retaining information on which entities are involved.

The entities in the BB3 and SeeDev subtasks are generally sets of full words but can be non-contiguous. Entities are stored as a set of associated words rather than a span of words. The GE4 task also contains entities that contain only partial words, for example, "PTEN" is tagged as an entity within "PTEN-deficient". A list of common prefixes and suffixes from the GE4 task is used to separate these words into two words so that the example would become "PTEN deficient". Furthermore, any annotation that divides a word that contains a hyphen or forward slash causes the word to be separate into two separate words.

For easier interoperability, the text parsing code was developed in Jython (Developers, 2008) (a version of Python that can load Java libraries, specifically the Stanford CoreNLP toolset). This Jython implementation is then able to export easily processed Python data structures. Due to incompatibility between Jython and various numerical libraries, a separate Python-only implementation loads the generated data structures for further processing

and classification.

3.2.3 Candidate generation

For all three classifications steps (entities, relations and modifications), the same machine learning framework is used. All possible candidates are generated for entities, relations or modifications. For relations, this means all pairs of entities are found (within a certain sentence range). For the training step, the candidates are associated with a known class (i.e. the type of relation), or the negative class if the candidate is not annotated in the training set. For testing, the classes are unknown. Candidates can contain one argument (for entity extraction and modification) or two arguments (for relation extraction). These arguments are stored as references to sentences and the indices of the associated words.

3.2.3.1 Entity extraction

Entity extraction aims to classify individual or sets of words as a certain type of entity, given a set of training cases. Entities may contain non-contiguous words. The set of all possible combinations of words that could compose an entity is too large for the classification system. Hence VERSE filters for only combinations of words that are identified as entities in the training set. This means that if the term "Lake Como" is annotated as a Habitat entity in the training set, any instance of "Lake Como" will be flagged as a candidate Habitat entity. However if a term (e.g. "the River Thames") never appears as an entity in the training set, it will be ignored for all test data.

3.2.3.2 Relation extraction

VERSE can predict relations between two entities, also known as binary relations. Candidates for each possible relation are generated for every pair of entities that are within a fixed sentence range. Hence when using the default sentence range of 0, only pairs of entities within the same sentence are analysed. VERSE can optionally filter pairs of entities using the expected types for a set of relations as shown in Figure 3.2.

Each candidate is linked with the locations of the two entities. If the two entities are already annotated to be in a relation, then they are labelled

Table 3.1: Overview of the various features that VERSE can use for classification

Feature Name	Target
unigrams	Entire Sentence
unigrams & parts-of-speech	Entire Sentence
bigrams	Entire Sentence
skipgrams	Entire Sentence
path edges type	Dependency Path
unigrams	Dependency Path
bigrams	Dependency Path
unigrams	Each Entity
unigrams & parts-of-speech	Each Entity
nearby path edge types	Each Entity
lemmas	Each Entity
entity types	Each Entity
unigrams of windows	Each Entity
is relation across sentences	N/A

with the corresponding class. Otherwise, the binary relation candidate is annotated with the negative class.

3.2.3.3 Modification extraction

VERSE supports modification of entities in the form of event modification but currently does not support modification of individual relations. A modification candidate is created for all entities that form the base of an event. These entities are often known as the triggers of the event. In the JSON format, these entities traditionally have IDs that start with "E". If a modification exists in the training set for that entity, the appropriate class is associated with it. Individual binary classifiers are generated for each modification type. This allows an event to be classified with more than one modification.

3.2.4 Features

For each generated candidate, a variety of features (controllable through a parameter) is calculated. The features focus on characteristics of the full sentence, dependency path or individual entities. The full-set is shown in Table 3.1. Each feature group, shown in the table, can be included or excluded with a binary flag. It should also be noted that a term frequency-inverse document frequency (TFIDF) transform is also an option for all bag-of-words based features.

3.2.4.1 Full sentence features

N-grams features (unigrams and bigrams) use a bag-of-words approach to count the word occurrences across the whole sentence. The words are transformed to lowercase but notably are not filtered for stop words. A version combining the individual words with part-of-speech information is also used. A bag-of-words vector is also generated for lemmas of all words in the sentence. Skip-gram-like features are generated using two words separated by a fixed window of words are also used to generate features. Hence the terms "regulation of EGFR" and "regulation with EGFR" would match the same features of "regulation * EGFR".

3.2.4.2 Dependency path features

The dependency path is the shortest path between the two entities in a dependency parse graph and has been shown to be important for relation extraction (Bunescu and Mooney, 2005). Features generated from the set of edges and nodes of the dependency graph include a unigrams and bigrams representation. The specific edge types in the dependency path are also captured with a bag-of-words vector. In order to give specific information about the location of the entity in the dependency path, the types of the edges leaving the entity nodes are recorded separately for each entity.

Interestingly an entity may span multiple nodes in the dependency graph. An example of a dependency path with the multi-word entities "coxiella burnetii" and "freshwater lakes" is shown in Figure 3.3. In this case, the minimal subgraph that connects all entity nodes in the graph is calculated. This problem was transformed into a minimal spanning tree problem as follows and solved using the NetworkX Python package (Hagberg et al.,

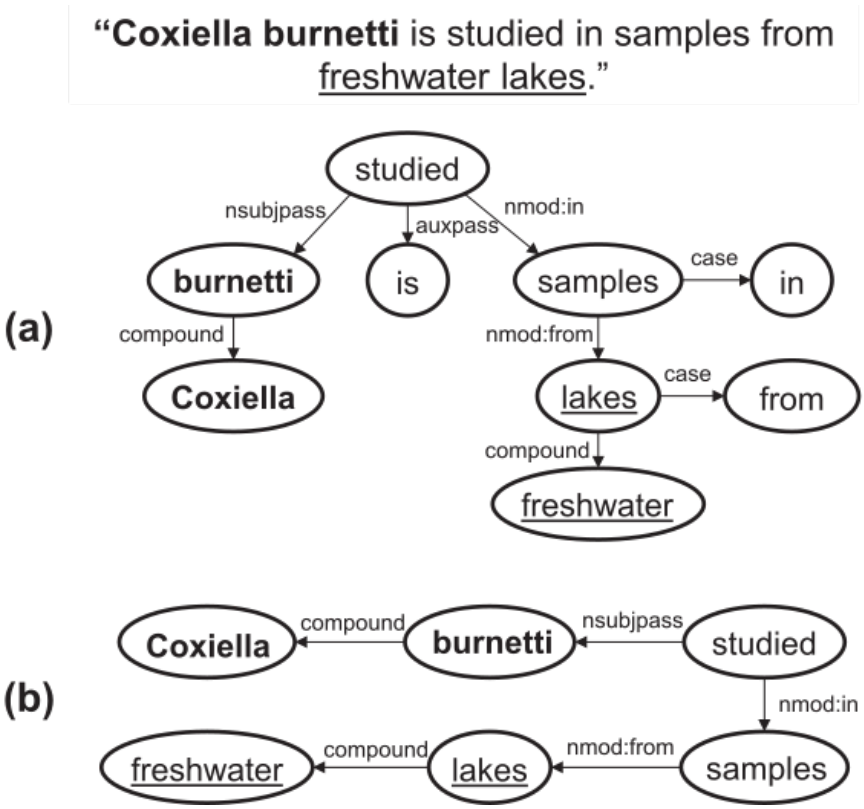


Figure 3.3: Dependency parsing of the shown sentence provides (a) the dependency graph of the full sentence which is then reduced to (b) the dependency path between the two multi-word terms. This is achieved by finding the subgraph which contains all entity nodes and the minimum number of additional nodes.

2008). The shortest paths through the graph were found for all pairs of entity nodes (nodes associated with the multi-word entities). The path distance between each pair was totalled and used to generate a new graph containing only the entity nodes. The minimal spanning tree was calculated and the associated edges recovered to generate the minimal subgraph. This approach would allow for a dependency path-like approach for relations between more than two entities.

3.2.4.3 Entity features

The individual entities are also used to generate specific features. Three different vectorised versions use a unigrams approach, a unigrams approach with parts-of-speech information and lemmas respectively. A one-hot vector approach is used to represent the type of each entity. Unigrams of words around each entity within a certain window size are also generated.

3.2.4.4 Multi-sentence and single entity features

VERSE is also capable of generating features for relations between two entities that are in different sentences. In this case, all sentence features are generated for both sentences together and no changes are made to the entity features.

The dependency path features are treated differently. The dependency path for each entity is created as the path from the entity to the root of the dependency graph, generally the main verb of the sentence. This then creates two separate paths, one per sentence and the features are generated in similar ways using these paths. Finally, a simple binary feature is generated for relation candidates that span multiple sentences.

For relation and modifications, candidates contain only a single argument. The dependency path is created in a similar manner to candidates of relations that span across sentences.

3.2.5 Classification

All candidates are vectorized using the same framework, whether for candidates with one or two arguments with minor changes. These vectorized candidates are then used for training a traditional classifier. The vectors

may be reduced using feature selection. Most importantly, the parameters used for the feature generation and classifier can easily be varied to find the optimal results. Classification uses the scikit-learn Python package (Pedregosa et al., 2011b).

3.2.5.1 Feature selection

VERSE implements optional feature selection using a chi-squared test on individual parameters against the class variable. The highest ranking features are then filtered based on the percentage of features desired.

3.2.5.2 Classifier parameters

Classification uses either a support vector machine (SVM) or logistic regression. When using the SVM, the linear kernel is used due to lower time complexity. The multi-class classification uses a one-vs-one approach. The additional parameters of the SVM that are optimised are the penalty parameter C, class weighting approach and whether to use the shrinking heuristic. The class weighting is important as the negative samples greatly outnumber the positive samples for most problems.

3.2.5.3 Stochastic parameter optimisation

VERSE allows adjustment of the various parameters including the set of features to generate, the classifier to use and the associated classification parameters. The optimisation strategy involves initially seeding 100 random parameter sets. After this initial set, the top 100 previous parameter sets are identified each iteration and one is randomly selected. This parameter set is then tweaked as follows. With a probability of 0.05, an individual parameter is changed. In order to avoid local maxima, an entirely new parameter set is generated with a probability of 0.1. For the subtasks, a 500 node cluster using Intel X5650s was used for optimisation runs.

The optimal parameters are determined for the entity extraction, relation extraction and each possible modification individually. In order to balance precision and recall equally at each stage, the F1-score is used.

3.2.6 Filtering

Final filtering is used to remove any predictions that do not fit into the task specification. Firstly all relations are checked to see that the types of the arguments are appropriate. Any entities that are not included in relations are removed. Finally, any modifications that do not have appropriate arguments or were associated with removed entities are also trimmed.

3.2.7 Evaluation

An evaluation system was created that generates recall, precision, and associated F1-scores for entities, relations and modifications. The system works conservatively and requires exact matches. It should be noted that our internal evaluation system gave similar but not exactly matching results to the online evaluation system for the BB3 and SeeDev subtasks.

K-fold cross-validation is used in association with this evaluation system to assess the success of the system. The entity, relation and modification extractors are trained separately. For the BB3 and SeeDev subtasks, two-fold cross-validation is used, using the provided split of training and development sets as the training sets for the first and second fold respectively. For the GE4 task, five-fold cross-validation is used. The average F1-score of the multiple folds is used as the metric of success.

3.3 Kindred Methods

The Kindred package was built as a follow up to the VERSE system. It is designed for generalizable relation extraction, is integrated with a wide variety of biomedical text mining resources and is distributed as a self-contained Python package for easy use.

Kindred is a Python package that builds upon the Stanford CoreNLP framework (Manning et al., 2014) and the scikit-learn machine learning library (Pedregosa et al., 2011a). The decision to build a package was based on the understanding that each text mining problem is different. It seemed more valuable to make the individual features of the relation extraction system available to the community than a bespoke tool that was designed to solve a fixed type of biomedical text mining problem. Python was selected due

to the excellent support for machine learning and the easy distribution of Python packages.

The ethos of the design is based on the scikit-learn API that allows complex operations to occur in very few lines of code, but also gives detailed control of the individual components. Individual computational units are encapsulated in separate classes to improve modularity and allow easier testing. Nevertheless, the main goal was to allow the user to download annotated data and build a relation extraction classifier in as few lines of code as possible.

3.3.1 Package development

The package has been developed for ease-of-use and reliability. The code for the package is hosted on Github. It was also developed using the continuous integration system Travis CI in order to improve the robustness of the tool. This allows regular tests to be run whenever code is committed to the repository. This will enable further development of Kindred and ensure that it continues to work with both Python 2 and Python 3. Coveralls and the Python coverage tool are used to evaluate code coverage and assist in test evaluation.

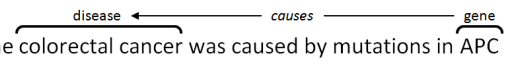
These approaches were in line with the recent recommendations on improving research software (Taschuk and Wilson, 2017). We hope these techniques will allow for and encourage others to make use of and contribute to the Kindred package.

3.3.2 Data Formats

As illustrated in Figure 3.4, Kindred accepts data in four different formats: the standoff format used by BioNLP Shared Tasks, the JSON format used by PubAnnotation, the BioC format (Comeau et al., 2013) and a simple tag format. The standoff format uses three files, a TXT file that contains the raw text, an A1 file that contains information on the tagged entities and an A2 file that contains information on the relations between the entities. The JSON, BioC and simple tag formats integrate this information into single files. The input text in each of these formats must have already been annotated for entities.

The simple tag format was implemented primarily for simple illustrations of Kindred and for easier testing purposes. It is parsed using an XML parser

Example Relation:



Standoff Format:

```
example.txt:
The colorectal cancer was caused by
mutations in APC
example.a1:
T1  disease 4 21 colorectal cancer
T2  gene 49 52   APC
example.a2:
E1  causes subj:T2 obj:T1
```

JSON Format:

```
{
  "text": "The colorectal cancer was
caused by mutations in APC",
  "denotations":
     [{"id":"T1", "obj":"disease",
       "span":{"begin":4,"end":21}},
      {"id":"T2", "obj":"gene",
       "span":{"begin":49,"end":52}}],
  "relations":
     [{"id":"R1","pred":"causes",
       "subj":"T2", "obj":"T1"}]
}
```

BioC Format:

```
<?xml version='1.0' encoding='UTF-8'?><!DOCTYPE
collection SYSTEM 'BioC.dtd'>
<collection>
  <document>
      <offset>0</offset>
      <text>The colorectal cancer was caused by
mutations in APC</text>
      <annotation id="T1">
        <infon key="type">disease</infon>
        <location offset="4" length="17"/>
        <text>colorectal cancer</text>
      </annotation>
      <annotation id="T2">
        <infon key="type">gene</infon>
        <location offset="49" length="3"/>
        <text>APC</text>
      </annotation>
      <relation id="R1">
        <infon key="type">causes</infon>
        <node refid="T2" role="subj"/>
        <node refid="T1" role="obj"/>
      </relation>
    </passage>
  </document>
</collection>
```

Simple Tag Format:

```
The <disease id="T1">colorectal cancer</disease> was caused by mutations in <gene
id="T2">APC</gene><relation type="causes" subj="T2" obj="T1" />
```

Figure 3.4: An example of a relation between two entities in the same sentence and the representations of the relation in four input/output formats that Kindred supports.

to identify all tags. A relation tag should contain a "type" attribute that denotes the relation type (e.g. causes). All other attributes are assumed to be arguments for the relation and their values should be IDs for entities in the same text. A non-relation tag is assumed to be describing an entity and should have an ID attribute that is used for associating relations.

3.3.3 Parsing and Candidate Building

The text data is loaded, and where possible, the annotations are checked for validity. In order to prepare the data for classification, the first step is sentence splitting and tokenization. We use the Stanford CoreNLP toolkit for this which is also used for dependency parsing for each sentence.

Once parsing has completed, the associated entity information must then be matched with the corresponding sentences. An entity can contain non-contiguous tokens as was the case for the BB3 event dataset in the BioNLP 2016 Shared Task. Therefore each token that overlaps with an annotation for an entity is linked to that entity.

Any relations that occur entirely within a sentence are associated with that sentence. The decision to focus on relations contained within sentence boundaries is based on the poor performance of relation extraction systems in the past. The VERSE tool explored predicting relations that spanned sentence boundaries in the BioNLP Shared Task and found that the false positive rate was too high. The sentence is also parsed to generate a dependency graph which is stored as a set of triples $(token_i, token_j, dependency_{ij})$ where $dependency_{ij}$ is the type of edge in the dependency graph between tokens i and j. The edge types use the Universal Dependencies format (Nivre et al., 2016).

Relation candidates are then created by finding every possible pair of entities within each sentence. The candidates that are annotated relations are stored with a class number for use in the multiclass classifier. The class zero denotes no relation. All other classes denote relations of specific types. The types of relations and therefore how many classes are required for the multiclass classifier are based on the training data provided to Kindred.

3.3.4 Vectorization

Each candidate is then vectorized in order to transform the tokenized sentence and set of entity information into a numerical vector that can be

processed using the scikit-learn classifiers. In order to keep Kindred simple and improve performance, it only generates a small set of features as outlined below.

- Entity types in the candidate relation
- Unigrams between entities
- Bigrams for the full sentence
- Edges in dependency path
- Edges in dependency path that are next to each entity.

For the entity type and edge relations, they are stored in a one-hot format. Entity specific features are created for each entity. For instance, if there are three relation types for relations between two arguments, then six binary features would be required to capture the entity types.

The unigrams and bigrams use a bag-of-words approach. Term-frequency inverse-document frequency (TF-IDF) is used for all bag-of-words based features. The dependency path, using the same method as VERSE, is calculated as the minimum spanning tree between the nodes in the dependency graph that are associated with the entities in the candidate relation.

3.3.5 Classification

Kindred has in-built support for the support vector machine (SVM) and logistic regression classifiers implemented in scikit-learn. By default, the SVM classifier is used with the vectorized candidate relations. The linear kernel has shown to give good performance and is substantially faster to train than alternative SVM kernels such as radial basis function or exponential.

The success of the LitWay and UniMelb entries to the SeeDev shared task suggested that individual classifiers for unique relation types may give improved performance. This may be due to the significant differences in complexity between different relation types. For instance, one relation type may require information from across the sentence for good classification, whereas another relation type may require only the neighboring word.

Using one classifier per relation type, instead of a single multiclass classifier, means that a relation candidate may be predicted to be multiple relation types. Depending on the dataset, this may be the appropriate decision as relations may overlap. Kindred offers this functionality of one classifier per

relation type. However, for the SeeDev dataset, we found that the best performance was actually through a single multiclass classifier.

3.3.6 Filtering

The predicted set of relations is then filtered using the associated relation type and types of the entities in the relation. Kindred uses the set of relations in the training data to infer the possible argument types for each relation.

3.3.7 Precision-recall tradeoff

The importance of precision and recall depends on the specific text mining problem. The BioNLP Shared Task has favored the F1-score, giving an equal weighting to precision and recall. Other text mining projects may prefer higher precision in order to avoid biocurators having to manually filter out spurious results. Alternatively, projects may require higher recall in order to not miss any possibly important results. Kindred gives the user the control of a threshold for making predictions. In this case, the logistic regression classifier is used as it allows for easier thresholding. This is because the underlying predicted values can be interpreted as probabilities. We found that logistic regression achieved performance very close to the SVM classifier. By selecting a higher threshold, the classifier will become more conservative, decrease the number of false positives and therefore improve precision at the cost of recall. By using cross-validation, the user can get an idea of the precision-recall tradeoff. The tradeoffs for the BB3 and SeeDev tasks are shown in Figure 3.5. This allows the user to select the appropriate threshold for their task.

3.3.8 Parameter optimization

TEES took a grid-search approach to parameter optimization and focused on the parameters of the SVM classifier. VERSE had a significantly larger selection of parameters and grid search was not computationally feasible so a stochastic approach was used. Both approaches are computationally expensive and generally need a computer cluster.

Kindred takes a much simpler approach to parameter optimization and can work out of the box with default values. To improve performance, the user can choose to do minor parameter optimization. The only parameter

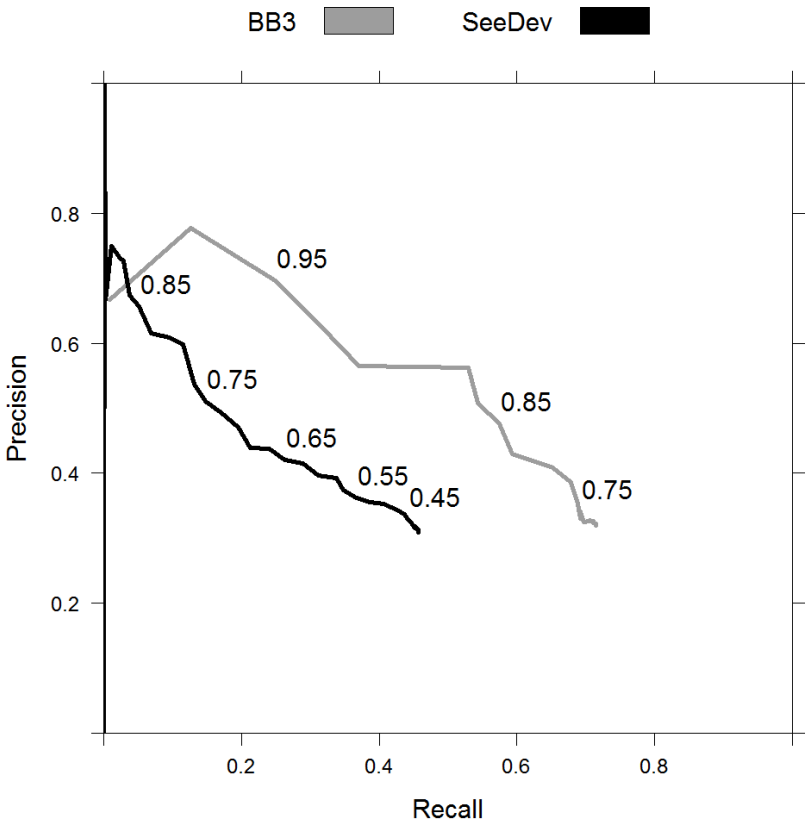


Figure 3.5: The precision-recall tradeoff when trained on the training set for the BB3 and SeeDev results and evaluating on the development set using different thresholds. The numbers shown on the plot are the thresholds.

optimized by Kindred is the exact set of features used for classification. This decision was made with the hypothesis that some relations potentially require words from across the sentence and other need only the information from the dependency parse.

The feature choice optimization uses a greedy algorithm. It calculates the F1-score using cross validation for each feature type. It then selects the best one and tries adding the remaining feature types to it. It continues growing the feature set until the cross-validated F1 score does not improve.

Figure 3.6 illustrates the process for the BB3 subtask using the training set and evaluating on the development set. At the first stage, the entity types feature is selected. This is understandable as the types of entity are highly predictive of whether a candidate relation is reasonable for a particular candidate type, e.g. two gene entities are unlikely to be associated in a 'IS_TREATMENT_FOR' relation. At the next stage, the unigrams between entities feature is selected. And on the third stage, no improvement is made. Hence for this dataset, two features are selected. We use this approach for the BB3 dataset but found that the default feature set performed best for the SeeDev dataset.

3.3.9 Dependencies

The main dependencies of Kindred are the scikit-learn machine learning library and the Stanford CoreNLP toolkit. Kindred will check for a locally running CoreNLP server and connect if possible. If none is found, then the CoreNLP archive file will be downloaded. After checking the SHA256 checksum to confirm the file integrity, it is extracted. It will then launch CoreNLP as a background process and wait until the toolkit is ready before proceeding to send parse requests to it. It also makes sure to kill the CoreNLP process when the Kindred package exits. Kindred also depends on the wget package for easy downloading of files, the IntervalTree Python package for identifying entity spans in text and NetworkX for generating the dependency path (Schult and Swart, 2008).

3.3.10 PubAnnotation integration

In order to make use of existing resources in the biomedical text mining community, Kindred integrates with PubAnnotation. This allows annotated text to be downloaded from PubAnnotation and used to train classifiers.

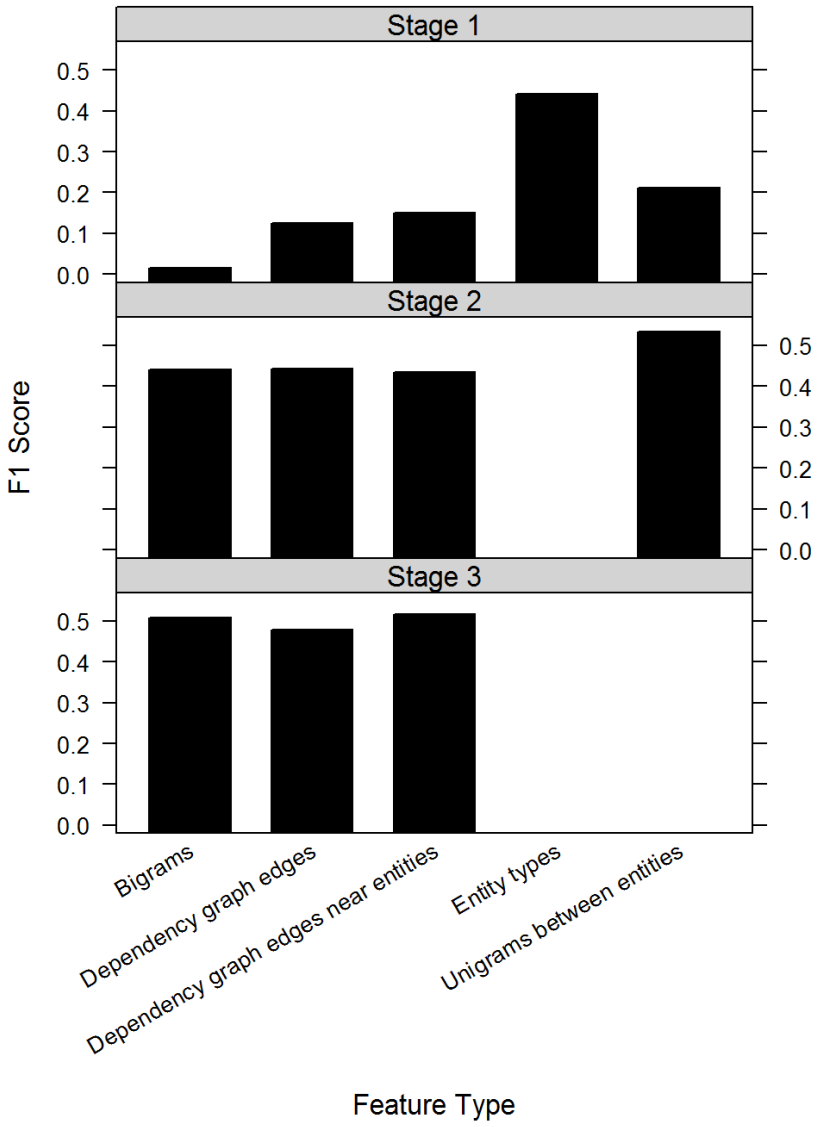


Figure 3.6: An illustration of the greedy approach to selecting feature types for the BB3 dataset.

The PubAnnotation platform provides a RESTful API that allows easy download of annotations from a given project. Kindred will initially download the listing of all available text sources with annotation for a given project. The listing is provided as a JSON data file. It will then download the complete set of texts with annotations.

3.3.11 PubTator integration

Kindred can also download a set of annotated PubMed abstracts that have already been annotated with named entities through the PubTator framework using the RESTful API. This requires the user to provide a set of PubMed IDs which are then requested from the PubTator server using the JSON data format. The same loader used for PubAnnotation data is then used for the PubTator data.

3.3.12 BioNLP Shared Task integration

Kindred gives easy access to the data from the most recent BioNLP Shared Task. By providing the name of the test and specific data set (e.g. training, development or testing), Kindred manages the download of the appropriate archive, unzipping and loading of the data. As with the CoreNLP dependency, the SHA256 checksum of the downloaded archive is checked before unzipping occurs.

3.3.13 API

One of the main goals of Kindred is to open up the internal functionality of a relation extraction system to other developers. The API is designed to give easy access to the different modules of Kindred that may be used independently. For instance, the candidate builder or vectorizer could easily be integrated with functionality from other Python packages, which would allow for other machine learning algorithms or deep learning techniques to be tested. Other parsers could easily be integrated and tested with the other parts of the Kindred in order to understand how the parser performance affects the overall performance of the system. We hope that this ease-of-use will encourage others to use Kindred as a baseline method for comparison in future research.

3.4 Results and discussion

The VERSE tool as described was applied to three subtasks: the BB3 event subtask, the SeeDev binary subtask and the GE4 subtask. The Kindred tool, which only focuses on relation extraction, is also compared to the top performing tools for the BB3 and SeeDev tasks.

3.4.1 Datasets

The BB3 event dataset provided by the BioNLP-ST 16 organizers contains a total of 146 documents (with 61, 34 and 51 documents in the training, development and test sets respectively). These documents are annotated with entities of the following types and associated total counts: bacteria (932), habitat (1,861) and geographical (110). Only a single relation type (Lives_In) is annotated which must be between a bacteria and habitat or a bacteria and a geographical entity.

The dataset for the SeeDev binary subtask contains 20 documents with a total of 7,082 annotated entities and 3,575 relations. There are 16 entity types and 22 relation types.

The GE4 dataset focuses on NFkB gene regulation and contains 20 documents. After filtering for duplicates and cleanup, it contains 13,012 annotated entities of 15 types. These entities are in 7,232 relations of 5 different types. It also contains 81 negation and 121 speculation modifications for events. Coreference data is also provided but was not used.

3.4.2 Cross-validated results

Both BB3 event and SeeDev binary subtasks required only relation extraction. VERSE was trained through cross-validation using the parameter optimising strategy and the optimal parameters are outlined in Table 3.2. Both tasks were split into training and development sets by the competition organisers. The training set contained roughly twice as many annotations as the development set. We used this existing split for the two-fold cross-validation. A linear kernel SVM was found to perform the best in both tasks. For both subtasks, relation candidates were generated ignoring the argument types as shown in Figure 3.2.

Table 3.2: Parameters used for BB3 and SeeDev subtasks

Parameter	BB3 event	SeeDev binary
Features	unigrams unigrams POS bigrams of dependency path unigrams of dependency path path edges types entity types entity lemmas entity unigrams POS path edges types near entities	unigrams unigrams POS path edges types path edges types near entities entity types
Feature Selection	No	Top 5%
Use TFIDF	Yes	Yes
Sentence Range	0	0
SVM Kernel	linear	linear
SVM C Parameter	0.3575	1.0 (default)
SVM Class Weights	Auto	5 for positive and 1 for negative
SVM Shrinking	No	No

Table 3.3: Cross-validated results of BB3 event subtask using optimal parameters

Metric	Fold 1	Fold 2	Average
Recall	0.552	0.610	0.581
Precision	0.469	0.582	0.526
F1-score	0.507	0.596	0.552

Table 3.4: Cross-validated results of SeeDev event subtask using optimal parameters

Metric	Fold 1	Fold 2	Average
Recall	0.363	0.386	0.375
Precision	0.261	0.246	0.254
F1-score	0.303	0.301	0.302

The classifiers for the two tasks use two very different sizes of feature vectors. The BB3 parameter set has a significant amount of repeated unigrams data, with unigrams for the dependency path and whole sentence with and without parts of speech. This parameter set also does not use feature selection, meaning that the feature vectors are very large (14,862 features). Meanwhile, the SeeDev parameters use feature selection to select the top 5% of features which reduces the feature vector from 7,140 features down to only 357. This size difference is very interesting and warrants further exploration of feature selection for other tasks.

Unfortunately, both classifiers performed best with a sentence range of zero, meaning that only relations within sentences could be detected. Tables 3.3 and 3.4 show the optimal cross-validated results that were found with these parameters. Notably, the F1-scores for the two folds of the SeeDev dataset are very similar, which is surprising given that the datasets are different sizes.

For the GE4 subtask, the cross-validation based optimisation strategy was used to find parameters for the entity, relation and modification extractions independently. Due to the larger dataset, filtering was applied to the argument types of relation candidates as shown in Figure 3.2. Table 3.5 outlines the resulting F1-scores from the five-fold cross-validations. As these extrac-

Table 3.5: Averaged cross-validated F1-score results of GE4 event subtask with entities, relations and modifications trained separately

Metric	Entities	Relations	Mods
Recall	0.703	0.695	0.374
Precision	0.897	0.736	0.212
F1-score	0.786	0.715	0.266

tors are trained separately, their performance in the full pipeline would be expected to be worse. This is explained as any errors during entity extraction are passed onto relation and modification extraction.

3.4.3 Competition results

The official results for the BB3 and SeeDev tasks are shown in Tables 3.6 and 3.7. Only VERSE competed in the competition as Kindred was developed at a later date. VERSE performed well in both tasks and was ranked first for the BB3 event subtask and third for the SeeDev binary subtask. The worse performance for the SeeDev dataset may be explained by the added complexity of many additional relation and entity types.

Table 3.8 shows the final results for the test set for the Genia Event subtask using the online evaluation tool. As expected, the F1-scores of the relation and modification extraction are lower for the full pipeline compared to the cross-validated results. Nevertheless, the performance is very reasonable given the more challenging dataset.

3.4.4 Multi-sentence analysis

29% of relations span sentence boundaries in the BB3 event dataset and 4% in the SeeDev dataset. Most relation extraction systems do not attempt to predict these multi-sentence relations. Given the higher proportion in the BB3 set, we use this dataset for further analysis of VERSE's ability to predict relations that span sentence boundaries. It should be noted that some of these relations may be artifacts due to false identification of sentence boundaries by the CoreNLP pipeline.

Table 3.6: Cross-validated results (Fold1/Fold2) and final test set results for VERSE and Kindred predictions in Bacteria Biotope (BB3) event subtask with test set results for the top three performing tools: VERSE, TurkuNLP and LIMSI.

Data	Precision	Recall	F1 Score
Fold 1	0.319	0.715	0.441
Fold 2	0.460	0.684	0.550
Kindred	0.579	0.443	0.502
VERSE	0.510	0.615	0.558
TurkuNLP	0.623	0.448	0.521
LIMSI	0.388	0.646	0.485

Table 3.7: Cross-validated results (Fold1/Fold2) and final test set results for Kindred predictions in Seed Development (SeeDev) binary subtask with test set results for the top three performing tools: LitWay, UniMelb and VERSE.

Data	Precision	Recall	F1 Score
Fold 1	0.333	0.411	0.368
Fold 2	0.255	0.393	0.309
Kindred	0.344	0.479	0.400
LitWay	0.417	0.448	0.432
UniMelb	0.345	0.386	0.364
VERSE	0.273	0.458	0.342

Table 3.8: Final reported results for GE4 subtask split into entity, relations and modifications results

Metric	Entities	Relations	Mods
Recall	0.71	0.23	0.11
Precision	0.94	0.60	0.38
F1-score	0.81	0.33	0.17

Using the optimal parameters for the BB3 problem, we analysed prediction results using different values for the sentence range parameter. The performance, shown in Figure 3.7, is similar for relations within the same sentence using different sentence range parameters. However, as the distance of the relation embiggens, the classifier predicts larger ratios of false positives to true positives. With sentence range = 3, the overall F1-score for the development set has dropped to 0.326 from 0.438 when sentence range = 1.

The classifier is limited by the small numbers of multi-sentence relations to use as a training set. With a suitable amount of data, it would be worthwhile exploring the use of separate classifiers for relations that are within sentences and those that span sentences as they likely depend on different features.

3.4.5 Error propagation in events pipeline

It should be noted that at each stage of the event extraction pipeline (Figure 3.1), additional errors can be introduced. If entities are not identified, then relations cannot be built upon them. And if entities or relations are missed, modifications cannot be predicted for them. At each stage, we targetted optimal F1-score with equal balance of precision and recall. An interesting future direction would be an exploration of different methods to reduce this, either targeting high recall (with lower precision) at each stage with a final cleanup method, or a unified approach that solves all three steps together.

3.4.6 Kindred

In order to show the efficacy of Kindred, we evaluate the performance on the BioNLP 2016 Shared Task data for the BB3 event extraction subtask and the SeeDev binary relation subtask. Parameter optimization was used for BB3 subtask but not for the SeeDev subtask which used the default set of feature types. Both tasks used a single multiclass classifier. Tables 3.6 and 3.7 shows both the cross-validated results using the provided training/development split as well as the final results for the test set.

The results are in line with the best performing tools in the shared task. It is to be expected that it does not achieve the best score in either task. VERSE, which achieved the best score in the BB3 subtask, utilized a computational cluster to test out different parameter settings for vectorization as well as classification. LitWay, the winner of the SeeDev subtask, used

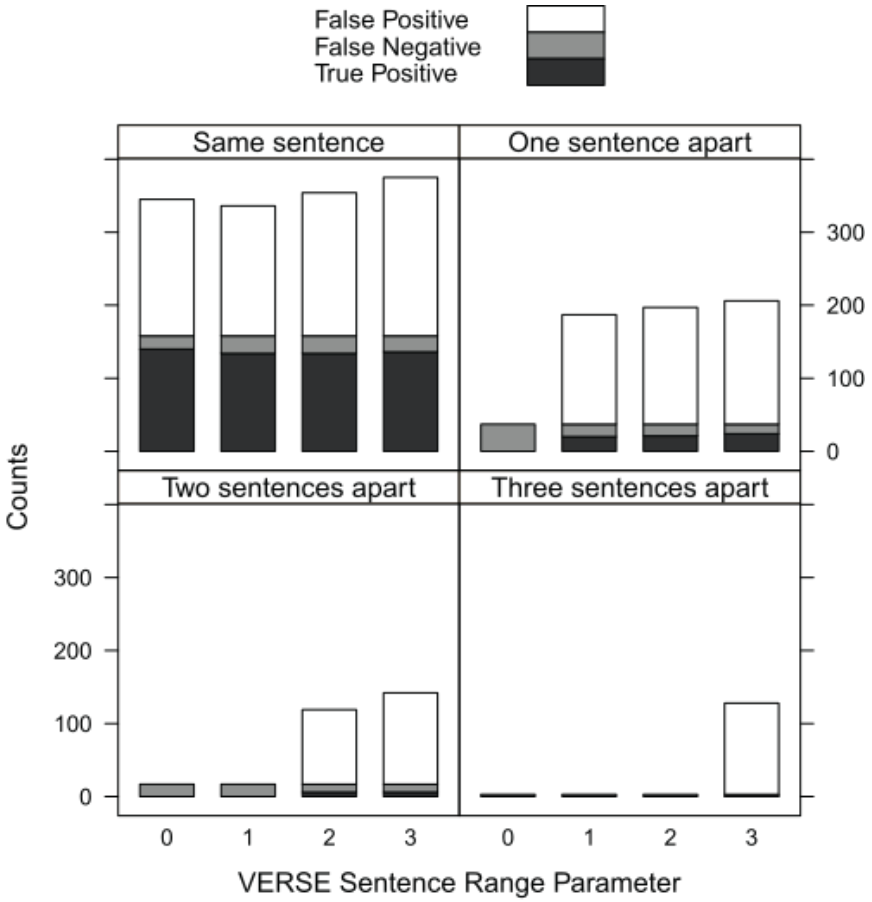


Figure 3.7: Analysis of performance on binary relations that cross sentence boundaries. The classifier was trained on the BB3 event training set and evaluated using the corresponding development set.

hand-crafted rules for a number of the relation types. Given the computational speed and simplicity of the system, Kindred is a valuable contribution to the community.

These results suggest several possible extensions of Kindred. Firstly, a hybrid system that mixes a vector-based classifier with some hand-crafted rules may improve system performance. This would need to be implemented to allow customization in order to support different biomedical tasks. Kindred is also geared towards PubMed abstract text, especially given the integration with PubTator. Using PubTator's API to annotate other text would allow Kindred to easily integrate other text sources, including full-text articles where possible. Given the open nature of the API, we hope that these improvements, if desired by the community, could be easily developed and tested.

Kindred has several weaknesses that we hope to improve. It does not properly handle entities that lie within tokens. For example, a token "HER2+", with "HER" annotated as a gene name, denotes a breast cancer subtype that is positive for the HER2 receptor. Kindred will currently associate the full token as a gene entity and will not properly deal the "+". This is not a concern for the BioNLP Shared Task problem but may become important in other text mining tasks.

3.5 Conclusion

We have presented VERSE, a full event extraction system that performed very well in the BioNLP 2016 Shared Task and its successor the Kindred Python package.

The VERSE system builds upon the success of previous systems, particularly TEES, in several important ways. It gives full control of the specific semantic features used to build the classifier. In combination with the stochastic optimisation strategy, this control has been shown to be important given the differing parameter sets found to be optimal for the different subtasks. Secondly, VERSE allows for feature selection which is important in reducing the size of the large sparse feature vectors and avoid overfitting. Lastly, VERSE can predict relations that span sentence boundaries, which is certain to be an important avenue of research for future relation extraction tasks. We hope that this tool will become a valuable asset in the biomedical text-mining community.

Kindred is designed for ease-of-use to encourage more researchers to test out relation extraction in their research. By integrating a selection of file formats and connecting to a set of existing resources including PubAnnotation and PubTator, Kindred will make the first steps for a researcher less cumbersome. We also hope that the codebase will allow researchers to build upon the methods to make further improvements in relation extraction research.

Chapter 4

A literature-mined resource for drivers, oncogenes and tumor suppressors in cancer

4.1 Introduction

As sequencing technology becomes more widely integrated into clinical practice, genomic data from cancer samples is increasingly being used to support clinical decision making as part of precision medicine efforts. Many initiatives use targeted panels that focus on well understood cancer genes, however more comprehensive approaches such as exome or whole genome sequencing that often uncover variants in genes of uncertain relevance to cancer are increasingly being employed. Interpreting individual cancer samples requires knowledge of which mutations are significant in cancer development. The importance of a particular mutation depends on the role of the associated gene and the specific cancer type. The terms "oncogene" and "tumor suppressor" are commonly used to denote genes (or aberrated forms) that respectively promote or inhibit the development of cancer. Genes of special significance to a particular cancer type or subtype are often described as "drivers". A deletion or loss-of-function mutation in a tumor suppressor gene associated with the cancer type of the sample is potentially an important event for this cancer. Furthermore, amplifications and gain-of function mutations in oncogenes, and any somatic activity in known driver genes may be valuable information in understanding the mutational landscape of a given cancer sample. This knowledge can then help select therapeutic options and improve our understanding of markers of resistance in the particular cancer type.

A variety of methods exist to identify a gene as a driver, oncogene or tumor suppressor given a large set of genomic data. Many methods use the

background mutation rate and gene lengths to calculate a p-value for the observed number of somatic events in a particular gene (Kristensen et al., 2014). Other studies use the presence of recurrent somatic deletions or low expression to deduce that a gene is a tumor suppressor (Cheng et al., 2017). In-vitro studies that examine the effect of gene knockdowns on the cancer's development are also used (Zender et al., 2008).

Structured databases with information about the role of different genes in cancer, specifically as drivers, oncogenes and tumor suppressors, are necessary for automated analysis of patient cancer genomes. The Cancer Genome Atlas (TCGA) project has provided a wealth of information on the genomic landscape of over 30 types of primary cancers (Weinstein et al., 2013). Data from TCGA (and other resources) are presented in the IntOGen resource to provide easy access to lists of driver genes (Gonzalez-Perez et al., 2013). The Cancer Gene Census has been curated using data from COSMIC to provide known oncogenes and tumor suppressors (Futreal et al., 2004) but faces the huge cost of manual curation. The Network of Cancer Genes (Ciccarelli et al., 2018) builds on top of the Cancer Gene Census and integrates a wide variety of additional contextual data including cancer types in which the genes are frequently mutated. Other resources that provide curated information about cancer genes include TSGene (Zhao et al., 2015) and OnGene (Liu et al., 2017) but do not match them with specific cancer types. There are also two other resources that are no longer accessible for unknown reasons (NCI Cancer Gene Index and MSKCC Cancer Genes database).

Text mining approaches can be used to automatically curate the role of genes in cancer, by identifying mentions of genes and cancer types, and extracting their relations from abstracts and full-text articles. Machine learning methods have shown great success in building protein protein interaction (PPI) networks using such data (Szklarczyk et al., 2016). We present CancerMine, a robust and regularly updated resource that describes drivers, oncogenes and tumor suppressors in all cancer types using the latest ontologies. By weighting gene roles by the number of supporting papers and using a high-precision classifier, we mitigate the noisy biomedical corpora and extract highly relevant structured knowledge.

4.2 Methods

4.2.1 Corpora Processing

PubMed abstracts and full-text articles from PubMed Central Open Access (PMCOA) subset and Author Manuscript Collection (PMCAMC) were downloaded from the NCBI FTP website using the PubRunner framework (paper in preparation - https://github.com/jakelever/pubrunner). They were then converted to BioC format using PubRunner's convert functionality. This strips out formatting tags and other metadata and retains the Unicode text of the title, abstract and for PMCOA, the full article. The source of the text (title, abstract, article) is also encoded.

4.2.2 Entity recognition

Lists of cancer types and gene names were built using a subset of the Disease Ontology (DO) and NCBI gene lists. These were complemented by matching to the Unified Medical Language System (UMLS). For cancer types, this was achieved using the associated ID in DO or through exact string matching on the DO item title. For gene names, the Entrez ID was used to match with UMLS IDs. The cancer type was then associated with a DO ID, and the gene names were associated with their HUGO gene name. These cancer and gene lists were then pruned with a manual list of stop-words with several custom additions for alternate spellings/acronyms of cancers. All cancer terms with less than four letters were removed except for a selected set of abbreviations, e.g. GBM for glioblastoma multiforme.

The corpus text was loaded in BioC format and processed using the Kindred Python package which, as of v2.0, uses the Spacy IO parser (described in Chapter 3). Using the tokenization, entities were identified through exact string matching against tokens. Longer entity names with more tokens were prioritised and removed from the sentence as entities were identified. Fusion terms (e.g. BCR-ABL1) were identified by finding gene names separated by a hyphen or slash. Non-fusions, which are mentions with multiple genes symbols that actually refer to a single gene (e.g.l HER2/neu), were then identified when two genes with matching HUGO IDs were attached and combined to be a single non-fusion gene entity. Genes mentioned in the context of pathways were also removed (e.g. MTOR pathway) using a list of pathway related keywords.

4.2.3 Sentence selection

After Kindred parsing, the sentences with tagged entities were searched for those containing at least one cancer type and at least one gene name. These sentences were then filtered using the terms "tumor suppress", "oncogen" and "driv" to enrich for sentences that were likely discussing these gene roles.

4.2.4 Annotation

From the complete set, 1,600 of the sentences were then randomly selected and output into the BioNLP Shared Task format for ingestion into an online annotation platform. This platform was then used by three expert annotators who are all PhD students actively engaged in precision cancer projects. The platform presents each possible pair of a gene and cancer and the user must annotate this as driving, oncogene and tumor suppressor. The first 100 sentences were used to help the users understand the system, evaluate initial inter-annotator agreement, and adjust the annotation guidelines (available at the Github repository). The results were then discarded and the complete 1,500 sentences were annotated by the first two annotators. The third annotator then annotated the sentences that the first two disagreed on. The inter-annotator agreement was calculated using the F1-score. A gold corpus was created using the majority vote of the annotations of the three annotators.

4.2.5 Relation extraction

To create a training and test split, 75% of the 1500 sentences were used as a training set and a Kindred relation classifier was trained with an underlying logistic regression model for all three gene roles (Driver, Oncogene and Tumor_Suppressor). The threshold was varied to generate the precision-recall curves with evaluation on the remaining 25% of sentences. With the selection of the optimal thresholds, a complete model was trained using all 1,500 sentences. This model was then applied to all sentences found in PubMed, PMCOA and PMCAMC that fit the sentence requirements. The associated gene and cancer type IDs were extracted, entity names were normalized and the specific sentence was extracted.

4.2.6 Web portal

The resulting cancer gene roles data were aggregated by the triples (gene, cancer, role) in order to count the number of citations supporting each cancer gene role. This information was then presented through tabular and chart form using a Shiny web application.

4.2.7 Resource comparisons

The data from the Cancer Gene Census (CGC), IntOGen, TS and ONGene resources were downloaded for comparison. HUGO gene IDs in CancerMine were mapped to Entrez gene IDs. CGC data was mapped to Disease Ontology cancer types using a combination of the cancer synonym list created for CancerMine and manual curation. Oncogenes and tumor suppressors were extracted using the presence of "oncogene" or "TSG" in the "Role in Cancer" column. The mapped CGC data was then compared against the set of oncogenes and tumor suppressors in CancerMine. IntOGen cancer types were manually mapped to corresponding Disease Ontology cancer types and compared against all of CancerMine. The TSGene and ONGene gene sets were compared against the CancerMine gene sets without an associated cancer type.

4.2.8 CancerMine profiles and TCGA analysis

For each cancer type, the citation counts for each gene role that were in the top 30 cancer genes were then log10-transformed and rescaled so that the most important gene had the value of 1 for each cancer type. Gene roles with values lower than 0.2 for all cancer types were trimmed. The top 30 cancer types and genes were then hierarchical clustered for the associated heatmap.

The open-access VarScan somatic mutation calls for the seven TCGA projects (BRCA,COAD,LIHC,PRAD,LGG,LUAD,STAD) were downloaded from the GDC Data Portal (https://portal.gdc.cancer.gov). They were filtered for mutations that contained a stop gain or were classified as probably damaging or deleterious by PolyPhen. Tumor suppressor specific CancerMine profiles were generated that used all tumor suppressors for each cancer type. The citation counts were again log10-transformed and rescaled to produce the CancerMine tumor suppressor profile. Each TCGA sample

was represented as a binary vector matching the filtered mutations. The dot-product of a sample vector and a CancerMine profile vector produced the sum of citation weightings and gave the score. For each sample, the score was calculated for all seven cancer types and the highest score was used to label the sample. A sample that did not contain tumor suppressor mutations associated with any of the seven profiles or could not be labelled unambigously was labelled as 'none'.

4.3 Results

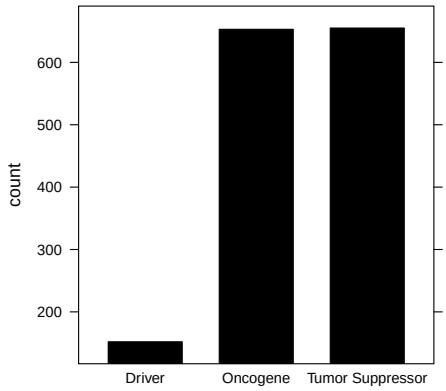
4.3.1 Role of 3,775 unique genes catalogued in 426 cancer types

The entire PubMed, PubMed Central Open Access subset (PMCOA) and PubMed Central Author Manuscript Collection (PMCAMC) corpora were processed to identify sentences that discuss a gene and cancer types within titles, abstracts and where accessible full text articles. By filtering for a customized set of keywords, these sentences were enriched for those likely discussing the genes' role and 1,500 randomly selected sentences were manually annotated by three expert annotators. Using a custom web interface and a well-defined annotation manual, the annotators tagged sentences that discussed one of three gene roles (driver, oncogene and tumor suppresser) with a mentioned type of cancer (Fig 4.1A). An example of a simple relation that was annotated as "Tumor Suppressor" annotation is: "DBC2 is a tumor suppressor gene linked to breast and lung cancers" (PMID: 17023000). A more complex example illustrates a negative relation: "KRAS mutations are frequent drivers of acquired resistance to cetuximab in colorectal cancers" (PMID:24065096). In this case, the KRAS mutations are drivers of drug resistance, and not of cancer development as required for annotation of driver relations.

With high inter-annotator agreement (Fig 4.1B), the data were split into 75%/25% training and test sets. A machine learning model was built for each of the three roles and precision-recall curves were generated (Fig 4.1C) using the test set. Receiver operating characteristic (ROC) curves were not used as the class balance for each relation was below 20%. A high threshold was selected for each gene role in order to provide high-precision prediction with the accepted trade-off of low recall (Fig 4.1D).

The trade-off of higher precision with lower recall was made based on the

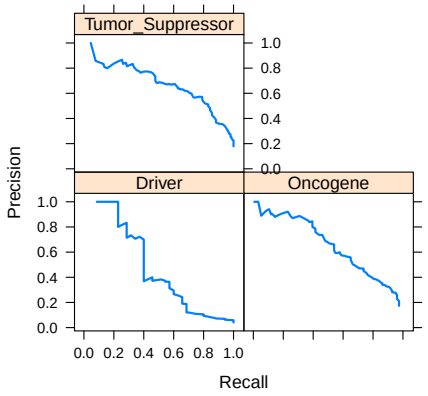
(a)



(b)

	Annotator 1	Annotator 2	Annotator 3
Annotator 1	1	0.803	0.791
Annotator 2	0.803	1	0.723
Annotator 3	0.791	0.723	1

(c)



(d)

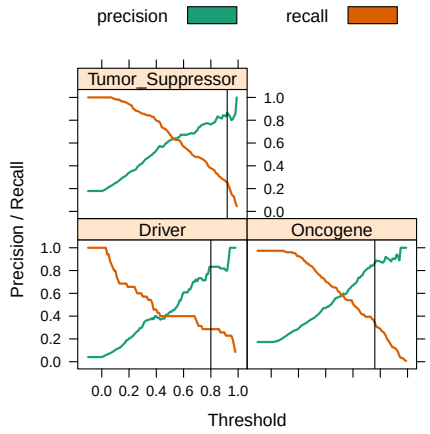


Figure 4.1: The supervised learning approach of CancerMine involves manual annotation by experts of sentences discussing cancer gene roles. Machine learning models are then trained and evaluated using this data set. (a) Manual text annotation of 1,500 randomly selected sentences containing genes and cancer types show a similar number of Oncogene and Tumor Suppressor annotations. (b) The inter-annotator agreement (measured using F1-score) was high between three expert annotators. (c) The precision recall curves show the trade-off of false positives versus false negatives. (d) Plotting the precision-recall data in relation to the threshold applied to the classifier's decision function provides a way to select a high-precision threshold.

hypothesis that there exists a large amount of redundancy within the published literature. The same idea is often stated multiple times in different papers in slightly different ways. Therefore, for frequently stated ideas, a method with lower recall would likely identify at least one occurrence. Nevertheless, we also distribute a version with thresholds of 0.5 for researchers who are willing to accept to a higher level of noise.

We apply the models to all sentences selected from PubMed abstracts and PMCOA/PMCAMC full-text articles, identifying 35,951 sentences from 26,767 unique papers that mention gene roles in cancer. We extract the unique gene/cancer pairs for each role (Fig 4.2A) and find that 3,775 genes and 426 cancer types are covered. These capture the commonly discussed cancer genes and types (Fig 4.2B/C) from a large variety of journals (Fig 4.2D). We provide a coverage of 21% (426/2,044) of the cancer types described in the Disease Ontology (Schriml et al., 2011) having at least one gene association. These results are made accessible through a web portal which can be explored through a gene or cancer-centric view. The resulting data are stored with Zenodo for versioning and download. This storage will provide the results in perpetuity. The results are licensed under the Creative Commons Public Domain (CC0) license to allow this data to be easily integrated with precision cancer workflows.

Our hypothesis of high levels of redundancy within the literature is supported by the frequent extraction of commonly-known gene roles such as ERBB2 as an oncogene in breast cancer (421 citations) and APC as a tumor suppressor in colorectal cancers (107 citations). On the other hand, a long tail exists of gene roles with only a single citation – 10,903 of 14,820 (73.6%) of extracted cancer gene roles (Fig 4.2E). For researchers that are accepting of a higher false positive rate, we provide an additional less stringent dataset using a lower prediction threshold and estimated average precision and recall of 0.5 and 0.6 respectively. The individual prediction score, akin to probabilities, are included so that users can further refine the results if needed.

4.3.2 60 novel putative tumor suppressors are published in literature each month

By examining the publication dates of the articles containing the mined cancer gene roles, we can see that the rate of published cancer gene roles is increasing over time (Fig 4.3A). In 2017, there were 6,851 mentions of cancer

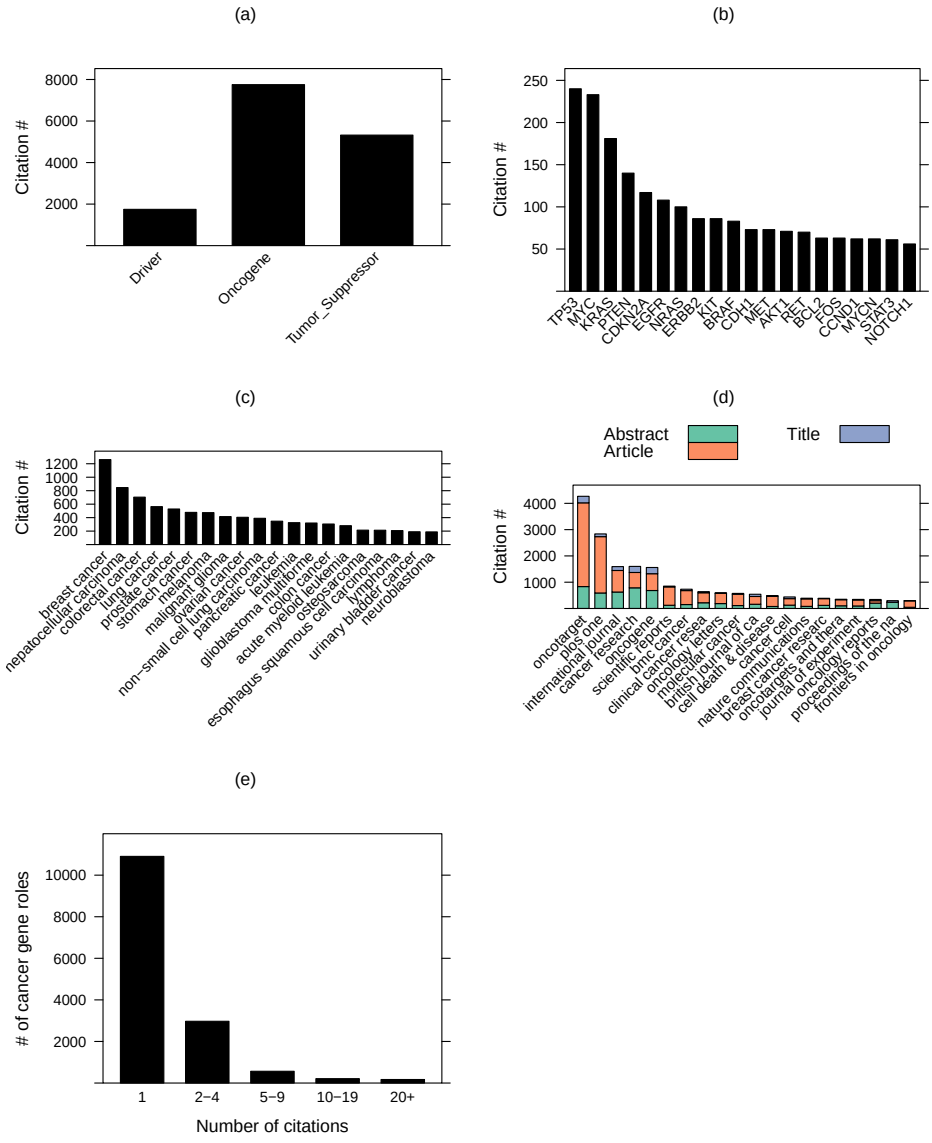


Figure 4.2: Overview of the cancer gene roles extracted from the complete corpora. (a) The counts of the three gene roles extracted. (b) and (c) show the most frequently extracted genes and cancer types in cancer gene roles. (d) The most frequent journal sources for cancer gene roles with the section of the paper highlighted by color. (e) illustrates a large number of cancer gene roles have only a single citation supporting it but that a large number (3917) have multiple citations.

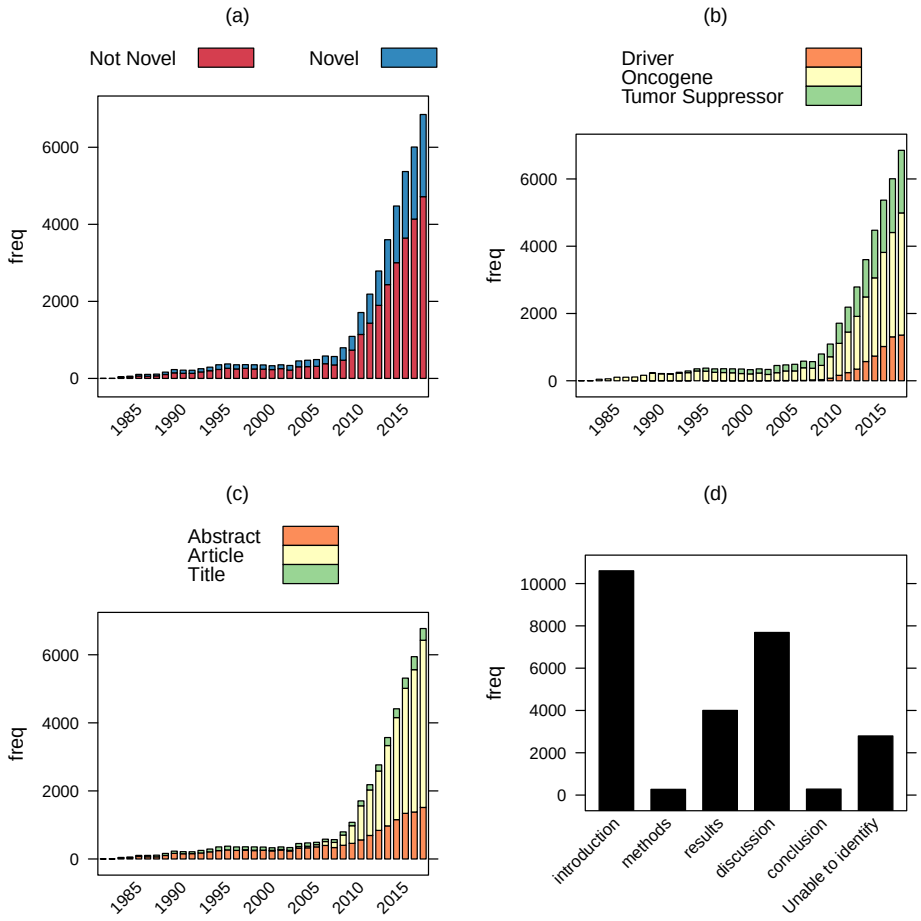


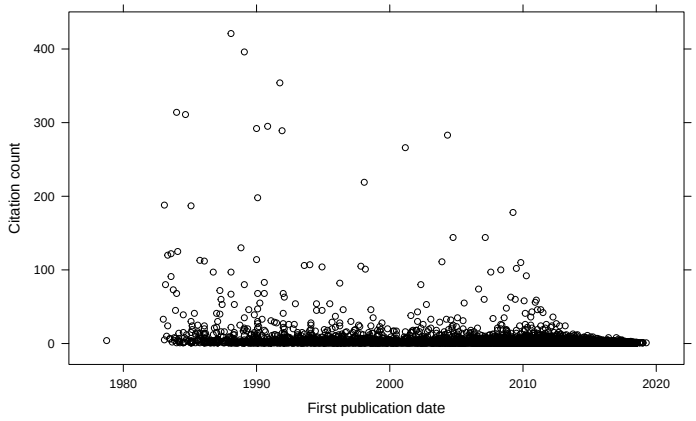
Figure 4.3: Examination of the sources of the extracted cancer gene roles with publication date. (a) More cancer gene roles are extracted each year but the relative proportion of novel roles remains roughly the same. (b) Roles extracted from older papers tend to focus on oncogenes, but mentions of driver genes have become more frequent since 2010. (c) The full text article is becoming a more important source of text mined data. (d) Different sections of the paper, particularly the Introduction and Discussion parts, are key sources of mentions of cancer gene roles (d).

gene roles in publications, translating to over ~571 each month. Approximately 69% of these are gene roles that have been published previously, but more importantly, the remaining 31% are novel. A breakdown by the role shows that oncogene and tumor suppressor gene mentions greatly outnumber driver genes. In 2017, 1,358, 3,632 and 1,861 genes were mentioned as drivers, oncogenes and tumor suppressors (Fig 4.3B). Combining this data, we find that there were, on average, 22 novel drivers, 96 novel oncogenes and 60 novel tumor suppressors described in literature each month. This emphasizes the need to keep these text mining results up-to-date at a frequency of less than a year. To this end, we have integrated the CancerMine resource with the PubRunner framework to execute intelligent updates once a month (paper forthcoming - https://github.com/jakelever/pubrunner).

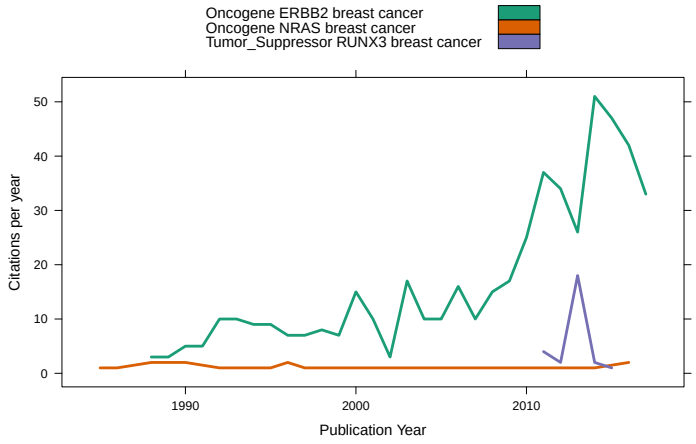
Unhindered access to the full-text of articles for text mining purposes remains a key challenge. A larger number of cancer gene role mentions are extracted from the full text (25,641) than from the abstract alone (15,291), with a smaller number extracted from the titles (4,150). As can be seen in Fig 4.3C, the number extracted from full text articles is increasingly dramatically over time. This is likely linked to the increasing number of publications included in the PubMed Central Open Access subset and Author Manuscript Collection. This strengthens the need for publishers to provide open access and for funding agencies to require publications in platforms that allow text mining. From the full text articles, we extract, where possible, the in-text location of the relationship captured within the paper (Fig 4.3D). Interestingly, a substantial number of the mentions are found in the Introduction section, suggesting that the cancer gene's role is usually discussed as background information and not a result of the paper. Knowing the subsection that a relationship is captured from can be valuable information for CancerMine users, since a user can then quickly ascertain if the discussed cancer role is prior knowledge or a likely result from the publication. This also highlights the important point that the scientific quality of a paper cannot be verified automatically by text mining technologies, since these methods rely on the statements made by the original author. Hence, any use of text-mined resources will always require users to access the original papers to evaluate the assertion of a gene's role in a particular cancer.

Cancer gene roles that are first mentioned at earlier timepoints have more time to accrue additional citations (Fig 4.4A). Thus, it is no surprise that while most cancer gene roles have less than 10 associated citations, those with very large citation counts tend to be published over 10 years ago. For instance, ERBB2's role as an oncogene in breast cancer is first extracted

(a)



(b)



(c)

Gene	Oncogenic in	Tumor Suppressive in
FOXP1	diffuse large B−cell lymphoma (6) hepatocellular carcinoma (6) ovarian cancer (4)	prostate cancer (4)
ID4	ovarian cancer (7)	prostate cancer (5)
MEN1	leukemia (5)	pituitary cancer (4)
NOTCH1	acute T cell leukemia (25) breast cancer (4) chronic lymphocytic leukemia (5) leukemia (5)	head and neck squamous cell carcinoma (14) lung small cell carcinoma (4) pancreatic ductal adenocarcinoma (4)
PTCH1	papillary thyroid carcinoma (8)	basal cell carcinoma (14)
RAB25	ovarian cancer (4)	colon cancer (6)
TCF3	leukemia (4)	hepatocellular carcinoma (4) stomach cancer (15)
WT1	leukemia (16) lung cancer (5)	childhood kidney neoplasm (9) kidney cancer (10) nephroblastoma (354)

Figure 4.4: (a) Cancer gene roles first discussed many years ago have a longer time to accrue further mentions. (b) Some cancer gene roles grow substantially in discussion while others fade away. (c) CancerMine can further validate the dual roles that some genes play as oncogenes and tumor suppressive. Citation counts are shown in parentheses.

from a publication in 1988 and has accumulated 421 citations that fit our extraction criteria in literature since then. However, there are some cancer gene roles that were first extracted from publications within the last decade but have already accrued a great number of additional mentions. For instance, KRAS driving non-small cell lung carcinoma is first extracted from a paper published in 2010, and already has 92 other papers mentioning this role since. Lastly, there are 691 cancer gene roles that are mentioned in literature before 2000, but are not extracted in papers after that period. The most frequently mentioned cancer gene role that reflects this pattern is MYC as a oncogene in cervix carcinoma, with 10 papers mentioning it before 2000 but no further citations afterwards.

With the knowledge of date of publication, we have gleaned a historical perspective on the gene relations captured in literature. Fig 4.4B summarizes three trends of citations that we observe, as exemplified by three gene associations with breast cancer. ERBB2 is an example of the small number of well established oncogenes that are more frequently discussed year upon year. NRAS in breast cancer exemplifies a gene that continues to be discussed in a single paper every few years, but has never gained importance in this cancer. RUNX3 has been discussed as a tumor suppressor in breast cancer in many papers in just the last few years. Its mechanism of action was elucidated after aggregated data from cell-line sequencing projects revealed its likely role as a tumor suppressor (Huang et al., 2012).

The cancer type is important when trying to understand the context of somatic mutations. This is underscored by examples such as NOTCH1. NOTCH1 is a commonly-cited gene that behaves as an oncogene in one cancer (acute T cell leukemia) and as a tumor suppressor in another (head and neck squamous cell carcinoma) (Radtke and Raj, 2003). We further validate CancerMine by querying the resource for the set of genes that are (i) strongly identified as a oncogene in at least one cancer type (>90% of >=4 citations) and (ii) strongly identified as a tumor suppressor in at least one other cancer type. This method successfully identifies NOTCH1 along with several other genes that are reported to play dual roles in different cancer types (Fig 4.4C).

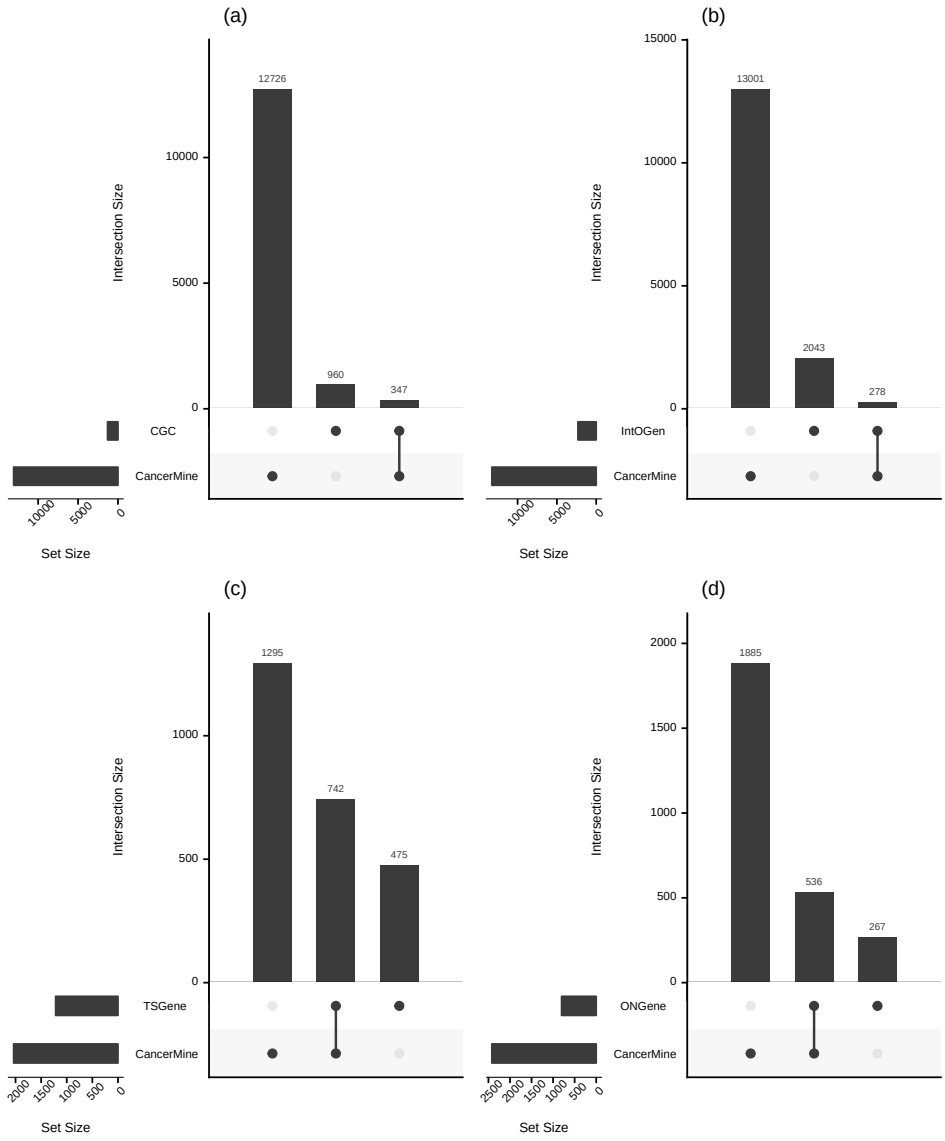


Figure 4.5: A comparison of CancerMine against resources that provide context for cancer genes. (a) The CancerMine resource contains substantially more cancer gene associations than the Cancer Gene Census resource. (b) Surprisingly few of the cancer gene associations are overlapping between the IntOGen resource and CancerMine . CancerMine overlaps substantially with the genes listed in the (c) TSGen and (d) ONGene resources.

4.3.3 Text mining provides voluminous complementary data to Cancer Gene Census

The Cancer Gene Census (CGC) (Futreal et al., 2004) provides manually curated information about cancer genes with mutation types and their roles in cancer. CancerMine contains information on 3,775 genes (compared to 554 in CGC) and 426 cancer types (compared to 201 in CGC). CancerMine overlaps with roughly a quarter of the oncogenes and tumor suppressors in the CGC when comparing specific cancer types (Fig 4.5A). When the CGC is compared to the less stringent CancerMine dataset, a further 202 cancer gene roles were found to match. This indicates that CGC contains curated information not easily captured using the sentence extraction method and that CancerMine represents an excellent complementary resource to work with CGC. Our resource also provides the sentence in which the gene role is discussed, and citations that link to the corresponding published literature are made available to help the user easily evaluate the evidence supporting the gene's role. CancerMine would be an excellent resource for prioritizing future curation of literature for resources such as CGC.

The IntOGen resource leverages a number of cancer sequencing projects, including the Cancer Genome Atlas (TCGA) to index genes inferred to contain driver mutations. A comparison of the genes with their cancer types in CancerMine shows surprising differences (Fig 4.5B). CancerMine includes a much larger set of genes but has little overlap with the IntOGen resource. This suggests that many of the genes identified through the projects included in IntOGen are not yet frequently discussed in the literature with respect to the specific cancer types in the IntOGen resource.

ONGene and TSGene2 provide lists of oncogenes and tumor suppressors. Unfortunately these gene names are not associated with specific cancer types which is an important aspect for precision oncology. When trying to differentiate between driving and passenger mutations, the lack of cancer type context would likely cause a high false positive rate. CancerMine contains ~67% of the genes in ONGene and ~61% of TSGene2, and contains substantially more genes than both resources (Fig 4.5C/D). These results lend more weight to the use of automated text mining approaches for the population of knowledge bases, since no curation is required to keep the resource up-to-date.

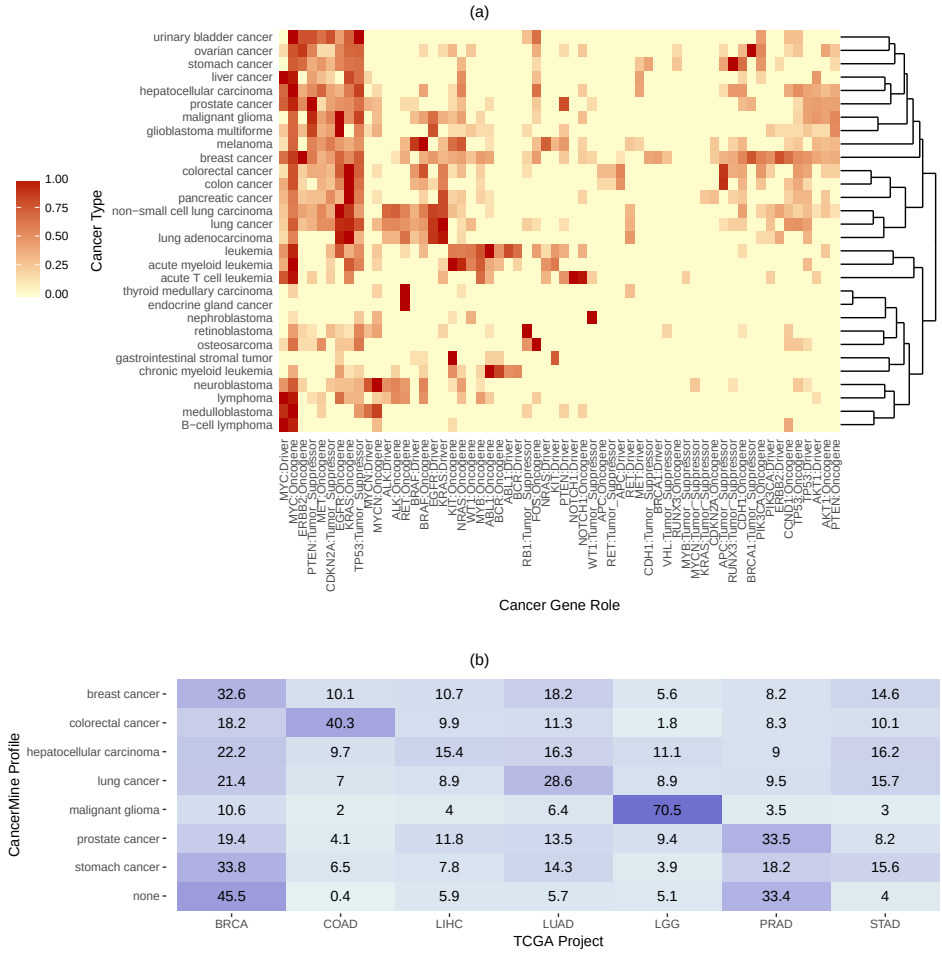


Figure 4.6: CancerMine data allows the creation of profiles for different cancer types using the number of citations as a weighting for each gene role. (a) The similarities between the top 30 cancer types in CancerMine are shown through hierarchical clustering of cancers types and genes using weights from the top 30 cancer gene roles. (b) All samples in seven TCGA projects are analysed for likely loss-of-function mutations compared with the CancerMine tumor suppressor profiles and matched with the closest profile. Percentages shown in each cell are the proportion of samples labelled with each CancerMine profile that are from the different TCGA projects. Samples that match no tumor suppressor in these profiles or are ambigious are assigned to none. The TCGA projects are breast cancer (BRCA), colorectal adenocarcinoma (COAD), liver hepatocellular carcinoma (LIHC), prostate adenocarcinoma (PRAD), low grade glioma (LGG), lung adenocarcinoma (LUAD) and stomach adenocarcinoma (STAD).

4.3.4 CancerMine provides insights into cancer similarities

Oncology often takes an organ-centric view of cancer types which is reflected by the numerous disease ontologies that exist for the categorization and nomenclature of cancer including the Disease Ontology used in this project. However, modern medicine is beginning to consider some cancers based purely on the genetic underpinnings, developing basket trials and approving treatment regimens based on genetic indications only (as shown with the successful approval of Pembrolizumab for PD-1 positive cancer patients). The CancerMine resource allows for the creation of a gene-centric view of cancers, by clustering cancers based on the role of different genes. A gene-centric view has the potential to reveal treatment regimes that could be transferred to other genetically similar cancer types. To allow for visualisation, we selected the top 30 cancers (based on citation count in CancerMine) and extracted the number of citations mentioning the role of the top 30 genes. This produces a profile for each cancer type showing the importance of each gene and its associated role. A heatmap that illustrates this for the top 30 cancer types and genes is shown in Figure 4.6A.

The clustering puts biologically similar or equivalent cancers together that are separate entities in the Disease Ontology. For example it groups colorectal with colon cancer and malignant glioma with glioblastoma multiforme. Some of these clusters also highlight known gene-cancer associations, for example, lung cancer, non-small cell lung carcinoma and lung adenocarinoma all cluster together, and are heavily associated with the KRAS and EGFR oncogenes. In fact, the strong cluster of genes on the left side separates cancers that are strongly associated with KRAS, EGFR, and TP53 (such as lung cancer) from those that are less so (such as thyroid medullary carcinoma). Put together, this approach is able to explain biological similarity of cancer types using shared gene associations. As an example, leukemia clusters closely with the more specific subtype, acute myeloid leukemia, and it is evident that this is driven by extracted associations of these cancers with MYC, ABL1 and many other genes. Several gene associations are noticeably low frequency compared to overall patterns, for instance KRAS in glioblastoma multiforme (GBM). While there are a small number of papers discussing KRAS in GBM, it is an infrequently discussed gene compared to EGFR and PTEN. Overall this visualisation presents an easy method to explore the similarities and differences between cancer types.

In order to validate the cancer genes identified in CancerMine, we compare results to somatic mutation data from the Cancer Genome Atlas (TCGA)

project. We hypothesis that the genes denoted to be tumor suppressors would likely be affected by loss-of-function mutations. Oncogenes may be affected by gain-of-function mutations which are harder to identify, hence our focus on tumor suppressors. Using CancerMine profiles based on tumor suppressor genes, we compare somatic calls for all samples with mutation data within seven TCGA projects. For each sample, we match the somatic calls against the set of CancerMine tumor profiles and sum the importance of the tumor suppressors found to be mutated. Figure 4.6B shows the percentages of top matches to each CancerMine profile. Six of the seven CancerMine profiles have their highest proportion matches with the corresponding TCGA project. Interestingly a large number of breast cancer and prostate cancer samples cannot be unambigiously labelled with one of the CancerMine profiles. For prostate cancer, roughly one third of the samples do not have any LoF mutations that match against any tumor suppressors for any of the seven types, suggesting that prostate cancer tumor suppressors are disabled through other mechanisms or that there are more tumor suppressors involved which have not been captured by CancerMine.

The glioma (LGG) result is the most prominent with 70.5% of TCGA LGG samples being most closely identified with the CancerMine malignant glioma profile. This is largely due to the high prevalence of IDH1 (390/503) mutations identified in the LGG cohort. While this data would not be enough for a tumor type classifier on its own, this results shows there is substantial signal that can be leveraged for interpreting the genomic data and could be combined further with other mutational data. This is underscored when examining breast cancer tumor suppressors with only a single citation, genes that are hypothetically not well known to be tumor suppressors in breast cancer. Seven of these genes (ARID1B, FGFR2, KDM5B, SPEN, TBX3, PRKDC and KMT2C) are mutated in at least 10 TCGA BRCA samples providing extra strength for the importance of these genes in breast cancer. In fact, the mechanism through which KMT2C inactivation drives tumorgenesis was recently elaborated in ER-positive breast cancer (Gala et al., 2018).

4.4 Discussion

This work contributes a much needed resource of known drivers, oncogenes and tumor suppressors in a wide variety of cancer types. The text mining approach taken is able to discern complicated descriptions of cancer gene

roles with a high level of precision. This provides for a continually updated resource with little need for human intervention. This generalizable method could extract other types of biological knowledge with only minor changes. However, there are several limitations to this approach that present interesting but challenging alleys for further investigation. Firstly, this method focuses on single sentence extraction due to the challenge of anaphora and coreference resolution across sentences. In Chapter 3, we showed that a high false positive rate occurs when identifying knowledge across multiple sentences. Our approach requires that authors discuss the gene name, role and cancer name all within the same sentence. This is a problem of writing style and probabilities that gets greatly diluted with the large number of publications processed. Furthermore our approach focuses on individual genes in isolation and is unable to capture complex interactions between cancer genes discussed in papers, e.g. mutual exclusivity. More of these complex relationships will likely be identified in future research and play a part in interpreting the somatic events in an individual cancer patient. Text mining approaches face growing challenges with extracting complex events like these, which may span multiple sentences or even paragraphs.

One important concept when interpreting CancerMine data is that our methodology does not force a definition of a driver, oncogene or tumor suppressor and relies on the assertion of individual authors. A decision was made to not extract discussion of genes frequently mutated in cancer. This was due to the acknowledged problem of huge genes (e.g. TTN) that frequently accrue many somatic mutations but likely don't play a part in cancer. Instead we rely on the authors' assertions of the role a gene plays in cancer. The level of evidence differs greatly as some assertions are based on intervential studies (e.g. knockdowns) while others use observational studies (e.g mutation frequency or expression experiments).

As has been noted, many attempts have been made to create a knowledge bases of this topic. Hosting the data through Zenodo and the code through Github provides a level of continuity that will guarantee that the project code and data stay accessible for the foreseeable future. Furthermore the PubRunner integration makes it easier to keep the results up-to-date. All data and analysis for this chapter is open source and documented. We hope others will explore this data in order to infer new knowledge of cancer types and their associated genes.

Chapter 5

Text-mining clinically relevant cancer biomarkers for curation into the CIViC database

5.1 Introduction

The ability to stratify patients into groups that are clinically related is an important step towards a personalized approach to cancer. Over time, a growing number of biomarkers have been developed in order to select patients who are more likely to respond to certain treatments. These biomarkers have also been valuable for prognostic purposes and for understanding the underlying disease biology by defining different molecular subtypes of cancers that should be treated in different ways (e.g. *ERBB2*/*ER*/*PR* testing in breast cancer (Onitilo et al., 2009)). Immunohistochemistry techniques are the primary approach for testing samples for diagnostic markers. (e.g. *CD15* and *CD30* for Hodgkin's disease (Rüdiger et al., 1998)). Recently, the lower cost and increasing speed of sequencing has allowed the DNA and RNA of individual patient samples to be characterized for clinical applications (Prasad et al., 2016). Throughout the world, this technology is beginning to inform clinician decisions on which treatments to use (Shrager and Tenenbaum, 2014). Such efforts are dependent on comprehensive and current understanding of the clinical relevance of variants. For example, the Personalized Oncogenomics project at the BC Cancer Agency identifies somatic events in the genome such as point mutations, copy number variations and large structural changes and, in conjunction with gene expression data, generates a clinical report to provide an 'omic picture of a patient's tumor (Jones et al., 2010).

The huge genomic variability in cancers means that each patient sample includes a huge number of new mutations, many of which have never been documented before (Chang et al., 2016). The phenotypic impact of most of these mutations is difficult to discern. This problem is exacerbated by the driver/passenger mutation paradigm where only a fraction of mutations are essential to the cancer (drivers) while many others have occurred through mutational processes that are irrelevant to the cancer and are deemed to have simply come along for the ride (passengers). An analyst trying to understand a new patient sample typically performs a literature review for each gene and specific variant. This is needed to understand its relevance in a cancer type, characterize the driver/passenger role of its observed mutations, and gauge the relevance for clinical decision making.

Several groups have built their own in-house knowledge bases which are developed as analysts examine increasing numbers of cancer patient samples. This tedious and largely redundant effort represents a substantial interpretation bottleneck impeding the progress of precision medicine (Good et al., 2014). To encourage a collaborative effort, the CIViC database (https://civicdb.org) was launched to provide a wiki-like editable online resource where edits and additions are moderated by experts in order to maintain high quality (Griffith et al., 2017). The resource provides information about clinically-relevant variants in cancer. Variants include protein-coding point mutations, copy number variations, epigenetic marks, gene fusions, aberrant expression levels and other 'omic events. It supports four types of biomarkers (also known as evidence types).

Diagnostic evidence items describe variants that can help a clinician diagnose or exclude a cancer. For instance, the JAK2 V617F mutation is a major diagnostic criterion for myeloproliferative neoplasms to identify polycythemia vera, essential thrombocythemia and primary myelofibrosis. Predictive evidence items describe variants that help predict drug sensitivity or response and are valuable in deciding further treatments. Predictive evidence items often explain mechanisms of resistance in patients who progressed on a drug treatment. For example, the ABL1 T315I missense mutation in the BCR-ABL fusion, predicts poor response to imatinib, a tyrosine kinase inhibitor that would otherwise effectively target BCR-ABL, in patients with chronic myeloid leukemia. Predisposing evidence items describe germline variants that increase the likelihood of developing a particular cancer, such as BRCA1 mutations for breast/ovarian cancer or RB1 mutations for retinoblastoma. Lastly, prognostic evidence items describe variants that predict survival outcome. As an example, colorectal cancers that harbor a KRAS mutation are

predicted to have worse survival.

CIViC presents this information in a human-readable text format consisting of an 'evidence statement' such as the sentence describing the ABL1 T315I mutation above together with data in a structured, programmatically accessible format. A CIViC 'evidence item' includes this statement, ontology-associated disease name (Schriml et al., 2011), evidence type as defined above, drug (if applicable), PubMed ID and other structured fields. Evidence items are manually curated and associated in the database with a specific gene (defined by Entrez Gene) and variant (defined by the curator).

Several other groups have created knowledge bases to aid clinical interpretation of cancer genomes. Many of these projects have joined the Variant Interpretation for Cancer Consortium (VICC, http://cancervariants.org/) to coordinate these efforts and have created a federated search mechanism to allow easier analysis across multiple knowledge bases (Wagner et al., 2018). The CIViC project is co-leading this effort along with OncoKB (Chakravarty et al., 2017), the Cancer Genome Interpreter (Tamborero et al., 2018), Precision Medicine Knowledge base (Huang et al., 2017), Molecular Match, JAX-Clinical Knowledge base (Patterson et al., 2016) and others.

Most of these projects focus on clinically-relevant genomic events, particularly point mutations, and provide associated clinical information tiered by different levels of evidence. Only CIViC includes RNA expression-based biomarkers. These may be of particular value for childhood cancers which are known to be 'genomically quiet', having accrued very few somatic mutations. Consequently, their clinical interpretation may rely more heavily on transcriptomic data (Adamson et al., 2014). Epigenomic biomarkers will also become more relevant as several cancer types are increasingly understood to be driven by epigenetic misregulation early in their development (Baylin and Ohm, 2006). For example, methylation of the MGMT promoter is a well known biomarker in brain tumors for sensitivity to the standard treatment, temozolomide (Hegi et al., 2005).

The literature on clinically relevant cancer mutations is growing at an extraordinary rate. For instance there were only 5 publications with BRAF V600E in title or abstract in PubMed in 2004 compared to 454 citations in 2017. In order to maintain a high quality and up-to-date knowledge base, a curation pipeline must be established. This typically involves a queue for papers, triaging those that should be curated and then assignment to a highly experienced curator. This prioritisation step is immensely important given the limited time of curators and the potentially vast number of papers

to be reviewed. Prioritisation must identify papers that contain knowledge that is of current relevance to users of the knowledge base. For instance, selecting papers for drugs that are no longer clinically approved would not be valuable to the knowledge base.

Text mining methods have become a common approach to help prioritise papers. These methods fall broadly into two main categories, information retrieval (IR) and information extraction (IE). IR methods focus on paper-level information and can take multiple forms. Complex search queries for specific terms or paper metadata (helped by the MeSH term annotations of papers in biomedicine) are common tools for curators. More advanced document clustering and topic modelling systems can use semi-supervised methods to predict whether a paper would be relevant for curation. Examples of this approach include the document clustering method used for the ORegAnno project (Aerts et al., 2008).

IE methods extract structured knowledge directly from the papers. This can take the form of entity recognition, by explicitly tagging mentions of biomedical concepts such as genes, drugs and diseases. A further step can involve relation extraction to understand the relationship discussed between tagged biomedical entities. This structured information can then be used to identify papers relevant for the knowledge base. IE methods are also used for automated knowledge base population without a manual curation step. For example, the mirTex knowledge base, which collates microRNA and their targets, uses automated relation extraction methods to populate the knowledge base (Li et al., 2015). Protein-protein interaction networks (such as STRING (Szklarczyk et al., 2016)) are often built using automatically generated knowledge bases.

The main objective of this project was to identify frequently discussed cancer biomarkers which fit the CIViC model but are not yet included in the CIViC knowledge base. We developed an IE-based method to extract key parts of the evidence item: cancer type, gene, drug (where applicable) and the specific evidence type from published literature. This allows us to count the number of mentions of specific evidence items in abstracts and full text articles and compare against the CIViC knowledge base. This chapter will present our methods to develop this resource, known as CIViCmine (http://bionlp.bcgsc.ca/civicmine/). This main contributions of this work are an approach for knowledge base construction that could be applied to many areas of biology and medicine, a machine learning method for extracting complicated relationships between four entity types, and extraction of

relationships across the largest possible publically accessible set of abstracts and full text articles. This resource, containing 70,655 biomarkers, is valuable to all cancer knowledge bases to aid their curation and also as a tool for precision cancer analysts searching for biomarkers not yet included in any other resource.

5.2 Methods

5.2.1 Corpora

The full PubMed and PubMed Central Open Access subset corpora was downloaded from the NCBI FTP website using the PubRunner infrastructure (Anekalla et al., 2017). These documents were converted to the BioC format for processing with the Kindred package (described in Chapter 3). HTML tags were stripped out and HTML special characters converted to Unicode. Metadata about the papers were retained including PubMed IDs, titles, journal information and publication date. Subsections of the paper were extracted using a customised set of acceptable section headers such as "Introduction", "Methods", "Results" and many synonyms of these. The corpora were downloaded in bulk in order to not overload the EUtils RESTFUL service that is offered by the NCBI. In order to avoid duplications of publications in PMCOA and PubMed, the PMIDs of all documents included in PMCOA were used to filter out abstracts from the PubMed corpus. The update files from PubMed were also processed to identify the latest version of each abstract to process.

5.2.2 Term Lists

Term lists were curated for genes, diseases and drugs based on several resources. The cancer list was curated from a section of the Disease Ontology (Schriml et al., 2011). All terms under the "cancer" (DOID:162) parent term were selected and filtered for unspecific names of cancer (e.g. "neoplasm" or "carcinoma"). These cancer types were then matched with synonyms from the Unified Medical Language System (UMLS) Metathesaurus (Bodenreider, 2004) (2017AB), either through existing external reference links in the Disease Ontology or through exact string-matching on the main entity names. The additional synonyms in the UMLS were then added through this link. The genes list was built from the Entrez gene list and complemented with

UMLS terms. Terms that overlapped with common words found in scientific literature (e.g. ice) were removed.

The drug list was curated from the WikiData resource (Vrandečić and Krötzsch, 2014). All Wikidata entities that are drug instances (Wikidata identifier: Q12140) were selected using a SPARQL query. The generic name, brand name and synonyms were extracted where possible. This link was complemented by a custom list of general drug categories (e.g. chemotherapy, tyrosine kinase inhibitors, etc) and a list of inhibitors built using the previously discussed gene list. This allowed for the extraction of terms such as "EGFR inhibitors". This was done because analysts are often interested in biomarkers associated with drug classes that target a specific gene, in addition to specific drugs.

All term lists were filtered with a stopwords list. This was based on the stopword list from the Natural Language Toolkit (Bird, 2006) and the most frequent 5,000 words found in the Corpus of Contempory American English (Davies, 2009) as well as custom set of terms. It was then merged with common words that occur as gene names (such as ICE).

A custom variant list was built that captured the main types of point mutations (e.g. loss of function), copy number variation (e.g. deletion), epigenetic marks (e.g. promoter methylation) and expression changes (e.g. low expression). These variants were complemented by a synonym list.

5.2.3 Entity extraction

The BioC corpora files were processed by the Kindred package. This NLP package used Stanford CoreNLP (Manning et al., 2014) for processing in the original published version (Lever and Jones, 2017). It was changed to Spacy (Honnibal and Johnson, 2015) for the improved Python bindings in version 2 for this project. This provided easier integration and execution on a cluster without running a Java subprocess. Spacy was used for sentence splitting, tokenization and dependency parsing of the corpora files.

Exact string matching was then used against the tokenized sentences to extract mentions of cancer types, genes, drugs and variants. Longer terms were prioritised during extraction so that "non small cell lung cancer" would be extracted instead of just "lung cancer". Variants were also extracted with a regular expression system for extracting protein coding point mutations (e.g. V600E).

Table 5.1: The five groups of search terms used to identify sentences that potentially discussed the four evidence types. Strings such as "sensitiv" are used to capture multiple words including "sensitive" and "sensitivity".

General	Diagnostic	Predictive	Predisposing	Prognostic
marker	diagnostic	sensitiv resistance efficacy predict	risk predispos	survival prognos DFS

Gene fusions (such as BCR-ABL1) were detected by identifying mentions of genes separated by a forward slash, hyphen or colon. If the two entities had no overlapping HUGO IDs, then it was flagged as a possible gene fusion and combined into a single entity. If there were overlapping IDs, it was deemed likely to be referring to the same gene. An example is HER2/neu which is frequently seen and refers to a single gene (ERBB2) and not a gene fusion.

Acronyms were also detected where possible by identifying terms in parentheses and checking the term before it, for instance "non-small cell lung carcinoma (NSCLC)". This was done to remove entity mistakes where possible. The acronym detection method takes the short form (the term in brackets) and iterates backwards through the long form (the term before brackets) looking for potential matches for each letter. If the long form and short form has overlapping associated ontology IDs, they likely refer to the same thing and can be combined, as in the example above. If only one of the long form or short form has an associated ontology ID, they are combined and assigned the associated ontology ID. If both long form and short form have ontology IDs but there is no overlap, the short form is disregarded as the long form has more likelihood of getting the specific term correct.

Gene mentions that are likely associated with signalling pathways and not specific genes (e.g. "MTOR signalling") are also removed using a simple pattern based on the words after the gene mention. One final post-processing step merges neighbouring terms with matching terms. So "HER2 neu" would be combined into one entity as the two terms (HER2 and neu) refer to the same gene.

5.2.4 Sentence selection

With all biomedical documents parsed and entities tagged, all sentences were selected that mention at least one gene, at least one cancer and at least one variant. A drug was not required as only one (Predictive) of the four evidence types involves a drug entity. These sentences were enriched by filtering with certain keywords that are strongly associated with the different evidence items. The full list and groupings of keywords are shown in Table 5.1. This grouping is done to make sure that each evidence type is represented reasonably equally in the training data. The General category with the keyword "marker" is included to catch additional sentences that discuss markers, which may relate to any of the four evidence types. Several of the keywords are stems in order to capture different forms of the word, e.g. prognosis or prognostic The acronym "DFS" which means "disease free survival" is also included as it was found in many sentences describing prognosis.

5.2.5 Annotation Platform

A web platform for simple relation annotation was built using Bootstrap (https://getbootstrap.com/). This allowed annotators to work using a variety of devices, including their smartphones. The annotation system could be loaded with a set of sentences with entity annotations stored in a separate file (also known as standoff annotations). When provided with a relation pattern, for example "Gene/Cancer", the system would search the input sentences and find all pairs of the given entity types in the same sentence. It would make sure that the two entities are not the same term, as in some sentences a token (or set of tokens) could be annotated as both a gene and a cancer, for instance "retinoblastoma". For a sentence with 2 genes and 2 cancer types, it would find all four possible pairs of gene and cancer type.

Each sentence, with all the possible candidate relations matching the relation pattern, would be presented to the user, one at a time (Fig 5.1). The user can then select various toggle buttons for the type of relation that these entities are part of. They can also use these to flag entity extraction errors or mark contentious sentences for discussion with other annotators.

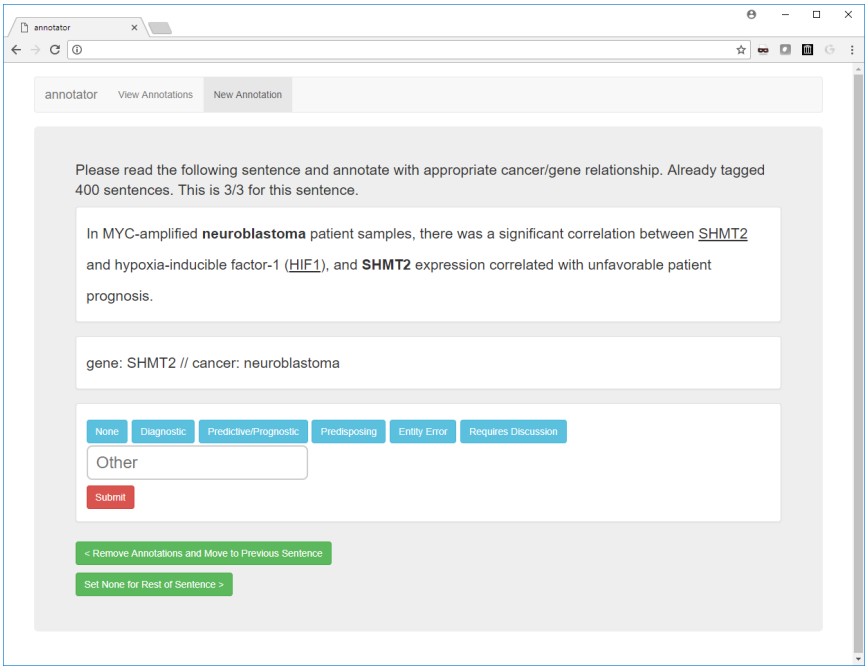


Figure 5.1: A screenshot of the annotation platform that allowed expert annotators to select the relation types for different candidate relations in all of the sentences. The example sentence shown would be tagged using "Predictve/Prognostic" as it describes a prognostic marker.

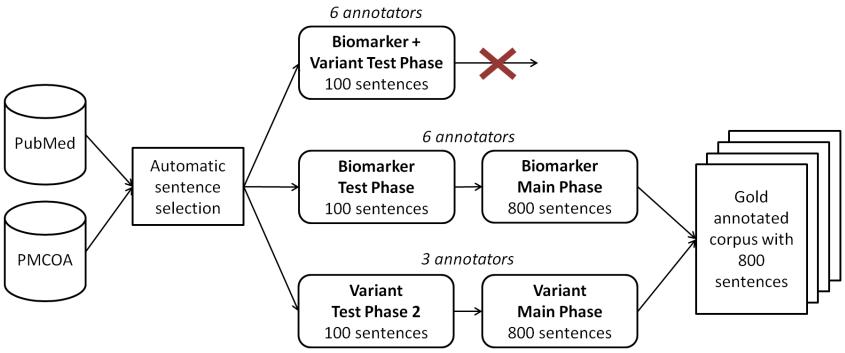


Figure 5.2: An overview of the annotation process. Sentences are identified from the literature that describe cancers, genes, variants and optionally drugs and then filtered using search terms. The first test phase tried complex annotation of biomarker and variants together but was unsuccessful. The annotation task was split into two separate tasks for biomarkers and variants separately. Each task had a test phase and then the main phase on the 800 sentences that were used to create the gold set.
(#fig:annotationOverview,)

5.2.6 Annotation

For the annotation stage (outlined in Fig **??**), an equal number of sentences were selected from each of the groups outlined in Table 5.1. This guaranteed coverage of all four evidence types as the prognostic type dominated the other groups. If this step was not done, 100 randomly selected sentences would only contain 2 (on average) from the Diagnostic group. However, this sampling provided poor coverage of sentences that describe specific point mutations. Many precision oncology projects only focus on point mutations and so a further requirement was that 50% of sentences for annotation include a specific point mutation. All together, this sampling provides better coverage of the different omic events and evidence types that were of interest. Special care is required when evaluating models built on this customized training set as an unweighted evaluation would not be representative of the real literature.

Sentences that contain many permutations of relationships (e.g. a sentence with 5 genes and 5 cancer types mentioned) were removed. An upper limit of 5 possible relations was enforced for each sentence. This was done with the knowledge that the subsequent relation extraction step would have a

greater false positive rate for sentences with very large number of possible relations. It was also done to make the annotation task more manageable. An annotation manual was constructed with examples of sentences that would and would not match the four evidence types. This was built in collaboration with CIViC curators. The annotation manual is available in our Github repository.

The annotation began with a test phase of 100 sentences that had poor annotator agreement and required a refinement of the annotation task outlined in this paragraph. The test phase allows the annotators to become accustomed to the annotation platform and make adjustments to the annotation manual to clarify misunderstandings. The first test phase (Biomarker + Variant) involved annotating sentences for ternary (gene, cancer, variant) or quaternary (gene, cancer, variant, drug) relationships. The ternary relationships included Diagnostic, Prognostic and Predisposing and the quaternary relationship was Predictive. A low F1-score inter-annotator agreement (average of 0.52) forced us to reconsider the annotation approach. This poor agreement was likely due to including variants within the annotations and provided a large combinatorial problem of exactly which entity mentions to include within a relationship. In order to simplify the problem, the task was split into two separate annotation tasks, the biomarker annotation and the variant annotation. The biomarker annotation involved binary (gene, cancer) and ternary (gene, cancer, drug) relations that described one of the evidence types. The Predictive and Prognostic evidence types were merged (as shown in Figure 2), to further reduce the annotation complexity. The Predictive/Prognostic annotations could be separated after tagging as relationships containing a drug would be Predictive and those without would be Prognostic. Any Prognostic relationship for a gene and cancer type that are in a Predictive relationship were removed. The variant annotation task (gene, variant) focused on whether a variant (e.g. deletion) was associated with a specific gene in the sentence.

With the redefined annotation task, six annotators were involved in biomarker annotation, all with knowledge of the CIViC platform and have experience interpreting patient cancer genome samples. Three annotators were involved in variant annotation, all with experience in cancer genomics. Both annotation tasks started with a new 100-sentence test phase to evaluate the redefined annotation tasks and resolve any ambiguity within the annotation manuals. Good inter-annotator agreement was achieved at this stage for both the biomarker annotation (average F1-score = 0.68) and variant annotation (average F1-score = 0.95). These 100 sentences

were discarded as they exhibited a learning curve as annotators become comfortable with the task.

(a)

	Annotator 2	Annotator 3
Annotator 1	0.74	0.73
Annotator 2	NA	0.74

(b)

	Annotator 2	Annotator 3
Annotator 1	0.78	0.85
Annotator 2	NA	0.79

(c)

	Annotator 2	Annotator 3
Annotator 1	0.96	0.96
Annotator 2	NA	0.96

Figure 5.3: The inter-annotator agreement for the main phase for 800 sentences, measured with F1-score, showed good agreement in the two sets of annotations for biomarkers (a) and (b) and very high agreement in the variant annotation task (c). The sentences from the multiple test phases are not included in these numbers and are discarded from the further analysis.

After a video-conference discussion, the annotation manuals were refined further. The main phase of biomarker annotation involved three annotators working on 400 sentences and the other three working on a different 400 sentences. Separately, three annotators worked on variant annotation with the 800 sentence set. Figure 5.3 shows the inter-annotator agreement for these tasks for the full 800 sentences. Each sentence is annotated by three annotators and a majority vote system is used to solve conflicting annotations. The biomarker and variant annotations are then merged to create the gold corpus of 800 sentences used for machine learning system.

5.2.7 Relation extraction

The sentences annotated with relations were then processed using the Kindred relation extraction Python package. Relation extraction models were built for all five of the relation types: the four evidence types (Diagnostic, Predictive, Predisposing and Prognostic) and one AssociatedVariant relation type. Three of the four evidence type relations are binary between a Gene entity and a Cancer entity. The AssociatedVariant relation type is also binary between a Gene entity and a Variant entity. The Predictive evidence

item type was ternary between a Gene, a Cancer Type and a Drug.

Most relation extraction systems focus on binary relations (Björne and Salakoski, 2013, Bui et al. (2013)) and use features based on the dependency path between those two entities. The recent BioNLP Shared Task 2016 series included a subtask for non-binary relations (i.e. relations between three or more entities) but no entries were received (Chaix et al., 2016). Relations between 2 or more entities are known as n-ary relations where $n \geq 2$. The Kindred relation extraction package, based on the VERSE relation extraction tool (described in Chapter 3) which won part of the BioNLP Shared Task 2016, was enhanced to allow prediction of n-ary relations. First, the candidate relation builder was adapted to search for relations of a fixed n which may be larger than 2. This meant that sentences with 5 non-overlapping tagged entities would generate 60 candidate relations with $n = 3$. These candidate relations would then be pruned by entity types. Hence, for the Predictive relation type (with $n = 3$), the first entity must be a Cancer Type, the second a Drug and the third a Gene. Two of the features used are based on the path through the dependency graph between the entities in the candidate relation. For relations with more than two entities, Kindred made use of a minimal spanning tree within the dependency graph.

The default Kindred features (outlined below) were then constructed for this subgraph and the associated entities and sentences. All features were represented with 1-hot vectors or bag-of-words representations.

- Entity types in the relation
- Unigrams between each pair of entities within the relation
- Bigrams of the entire sentence
- All edge types within the minimal spanning tree of the dependency graph that links all entity nodes
- Edge types of edges that are attached to entity nodes within the dependency graph

During training, candidate relations are generated with matching n-ary to the training set. Those candidate relations that match a training example are flagged as positive examples with all others as negative. These candidate relations are vectorized and a logistic regression classifier is trained against them. The logistic regression classifier outputs an interpretable score akin to a probability for each relation, which was later used for filtering. Kindred also supports a Support Vector Machine classifier (SVM) or can be extended

Table 5.2: Number of annotations in the training and test sets

Annotation	Train	Test
AssociatedVariant	768	270
Diagnostic	156	62
Predictive	147	43
Predisposing	125	57
Prognostic	232	88

with any classifier from the scikit-learn package (Pedregosa et al., 2011a). The logistic regression classifier was more amenable for adjustment of the precision-recall tradeoff.

For generation of the knowledge base, the four evidence type relations were predicted first which provided relations including a Gene. The Associated-Variant relation was then predicted and attached to any existing evidence type relation that included that gene.

5.2.8 Evaluation

With the understanding that the annotated sentences were selected randomly from customised subsets and not randomly from the full population, care was taken in the evaluation process.

First, the annotated set of 800 sentences was split 75%/25% into a training and test set that had similar proportions of the four evidence types (Table 5.2). Each sentence was then tracked with the group it was selected from (Table 5.1). Each group has an associated weight based on the proportion of the entire population of possible sentences that it represents. Hence, the Prognosis group, which dominates the others, has the largest weight. When comparing predictions against the test set, the weighting associated with each group was then used to adjust the confusion matrix values. The goal of this weighting scheme was to provide performance metrics which would be representative for randomly selected sentences from the literature and not for the customised training set.

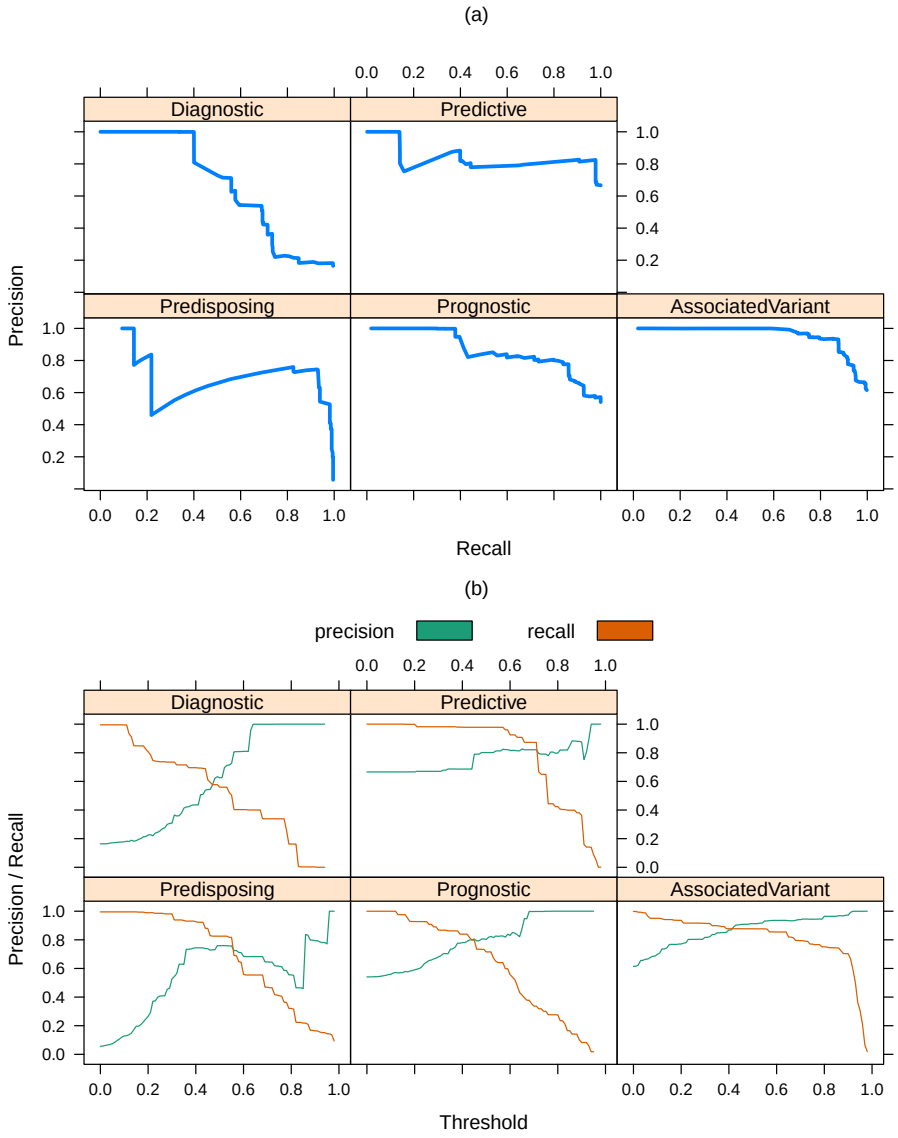


Figure 5.4: (a) The precision-recall curves illustrate the performance of the five relation extraction models built for the four evidence types and the associated variant prediction. (b) This same data can be visualised in terms of the threshold values on the logistic regression to select the appropriate value for high precision with reasonable recall.

Table 5.3: The selected thresholds for each relation type with the high precision and lower recall trade-off.

Extracted Relation	Threshold	Precision	Recall
AssociatedVariant	0.70	0.941	0.794
Diagnostic	0.63	0.957	0.400
Predictive	0.93	0.891	0.141
Predisposing	0.86	0.837	0.218
Prognostic	0.65	0.878	0.414

5.2.9 Precision-recall Tradeoff

Figure 5.4a shows precision recall curves for all five of the relation types. The Diagnostic and Predisposing tasks are obviously the most challenging for the classifier. This same data can be visualised using the threshold values used against the output of the logistic regression for each metric (Fig 5.4b).

In order to provide a high quality resource, we decided on a trade off of high precision with low recall. We hypothesised that the most commonly discussed cancer biomarkers, which are the overall goal of this project, would appear in many papers using different wording. These frequently mentioned biomarkers would then be likely picked up even with lower recall. This also reduces the burden on CIViC curators to sift through false positives. With this, we selected thresholds that would give as close to 0.9 precision given the precision-recall curves for the four evidence types. We require a higher precision for the variant annotation (0.94). The thresholds and associated precision recall tradeoffs are shown for all five extracted relations in Table 5.3.

5.2.10 Application to PubMed and PMCOA

With the thresholds selected, the final models were applied to all sentences extracted from PubMed and PMCOA. This is a reasonably large computational problem and was tasked to the compute cluster at the Genome Sciences Centre.

In order to manage this compute and provide infrastructure for easy updating with new publications in PubMed and PMCOA, we made use of the updated Pubrunner infrastructure (paper in preparation - https://github.

com/jakelever/pubrunner). This allows for easy distribution of the work across a compute cluster. The resulting data was then pushed to Zenodo (https://zenodo.org/) for perpetual and public hosting. The data is released with a Creative Commons Public Domain (CC0) license so that other groups can easily make use of it.

5.2.11 CIViC Matching

In order to make comparisons with CIViC, we downloaded the nightly data file from CIViC (https://civicdb.org/releases) and matched evidence items against each other. The evidence type and IDs for genes and cancers were used for matching. Direct string matching was used to compare drug names for Predictive biomarkers. The exact variant was not used for comparison in order to find a genes that contain any biomarkers that match between the two resources.

Some mismatches occurred with drug names. For example, CIViCmine may capture information about the drug family while CIViC contains information on specific drugs, or a list of drugs. Another challenge with matching with CIViCmine is related to the similarity of cancer types in the Disease Ontology. There are several pairs of similar cancers types that are used interchangably by some researchers and not by others, e.g. stomach cancer and stomach carcinoma. CIViC may contain a biomarker for stomach cancer and CIViCmine matches all the other details except it relates it to stomach carcinoma.

5.2.12 User interface

In order to make the data easily explorable, we provide a Shiny based front-end (Fig **??**) (RStudio, Inc, 2013). This shows a list of biomarkers which can be filtered by the Evidence Type, Gene, Cancer Type, Drug and Variant. In order to help prioritize the biomarkers, we use the number of unique papers that the variants are mentioned in as a metric. By default, the listed biomarkers are shown with the highest citation count first. Whether the biomarker is found in CIViC is also shown as a column and is an additional filter. This allows CIViC curators to quickly navigate to biomarkers not currently discussed in CIViC and triage them efficiently.

With filters selected, the user is presented with pie-charts that illustrate the representation of different cancer types, genes and drugs. When the user

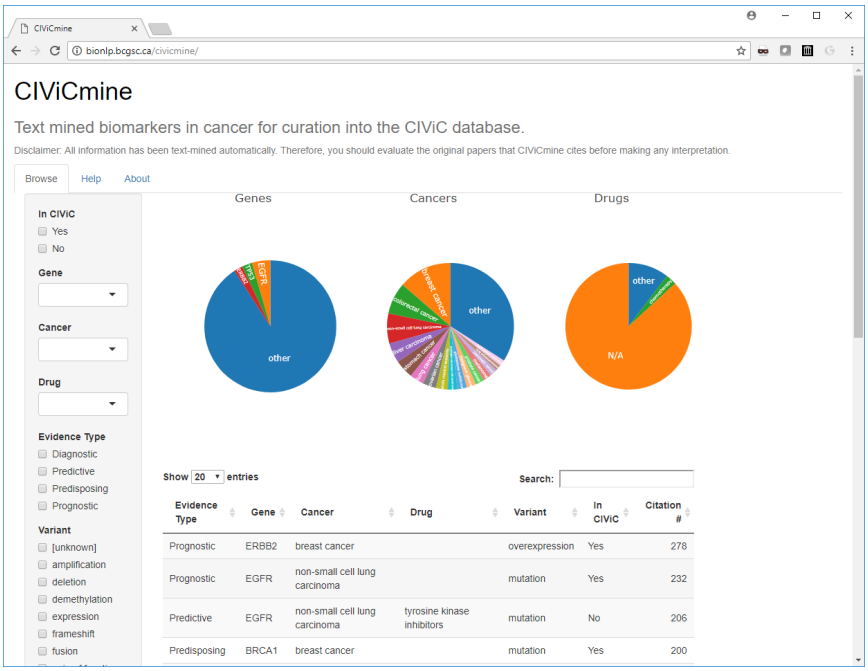


Figure 5.5: A Shiny-based web interface allows for easy exploration of the CIViCmine biomarkers with filters and overview piecharts. A main table shows the list of biomarkers and links to a subsequent table showing the list of supporting sentences.
(#fig:shiny,)

clicks on a particular biomarker, an additional table is populated with the citation information. This includes the journal, publication year, section of the publication (e.g. title, abstract or main body), subsection (if cited from the main body) and the actual text of the sentence. This table can further be searched and sorted, for example to look for older citations or citations from a particular journal. The PubMed ID is also provided with a link to the citation on PubMed.

5.3 Results

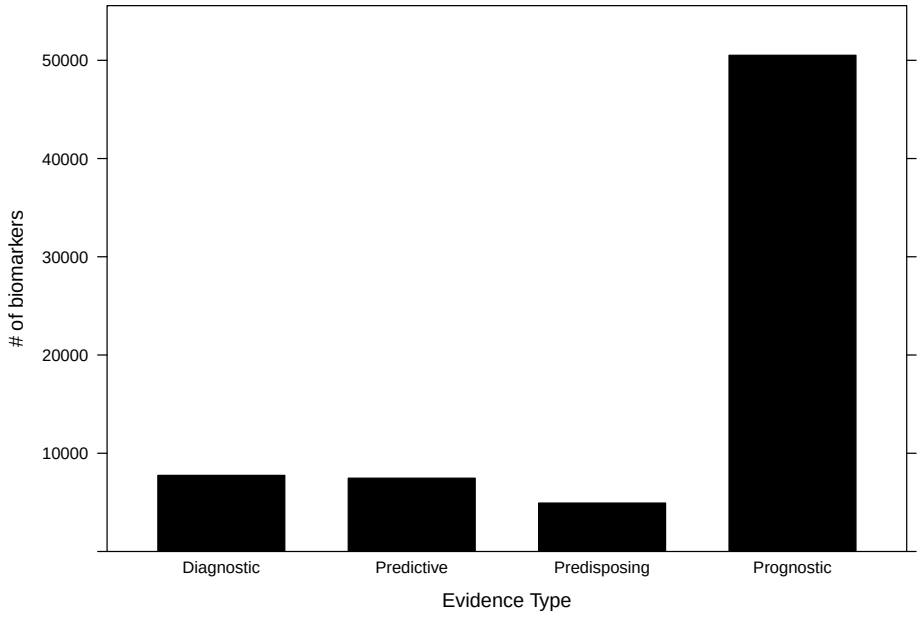


Figure 5.6: The entirety of PubMed and PubMed Central Open Access subset were processed to extract the four different evidence types shown.

From the full PubMed and PMCOA corpus, we extracted 70,655 biomarkers with a breakdown into the four types (Figure 5.6). As expected, there are many more Prognostic evidence items than the other three types. Table 5.4 outlines examples of all four of these evidence types. 34.9% of sentences (33,491/95,871) contain more than one evidence item, such as the Predictive example which relates EGFR as a predictive marker in NSCLC to both

Table 5.4: Four example sentences for the four evidence types extracted by CIViCmine. The associated PubMed IDs are also shown for reference.

Type	PMID	Sentence
Diagnostic	29214759	JAK2 V617F is the most common mutation in myeloproliferative neoplasms (MPNs) and is a major diagnostic criterion.
Predictive	28456787	In non-small cell lung cancer (NSCLC) driver mutations of EGFR are positive predictive biomarkers for efficacy of erlotinib and gefitinib.
Predisposing	28222693	Our study suggests that one BRCA1 variant may be associated with increased risk of breast cancer.
Prognostic	28469333	Overexpression of Her2 in breast cancer is a key feature of pathobiology of the disease and is associated with poor prognosis.

erlotinib and gefitinib. In total, we extracted 153,435 mentions of biomarkers from 54,274 unique papers. These biomarkers relate to 6,591 genes, 510 cancer types and 334 drugs.

EGFR and TP53 stand out as the most frequently extracted genes in different evidence items (Fig 5.7a). Over 50% of the EGFR evidence items are associated with lung cancer or non-small cell lung carcinoma (NSCLC). CDKN2A has a larger proportion of diagnostic biomarkers associated with it than most of the other genes in the top 20. CDKN2A expression is a well-established marker for distinguishing HPV+ versus HPV- cervical cancers. Its expression or methylation are discussed as diagnostic biomarkers in a variety of other cancer types including colorectal cancer and stomach cancer.

Breast cancer is, by far, the most frequently discussed cancer type (Fig 5.7b). A number of the associated biomarkers focus on predisposition, as breast cancer has one of the strongest hereditary components associated with germline mutations in BRCA1 and BRCA2. NSCLC shows the largest relative number of predictive biomarkers, consistent with the previous figure showing the importance of EGFR.

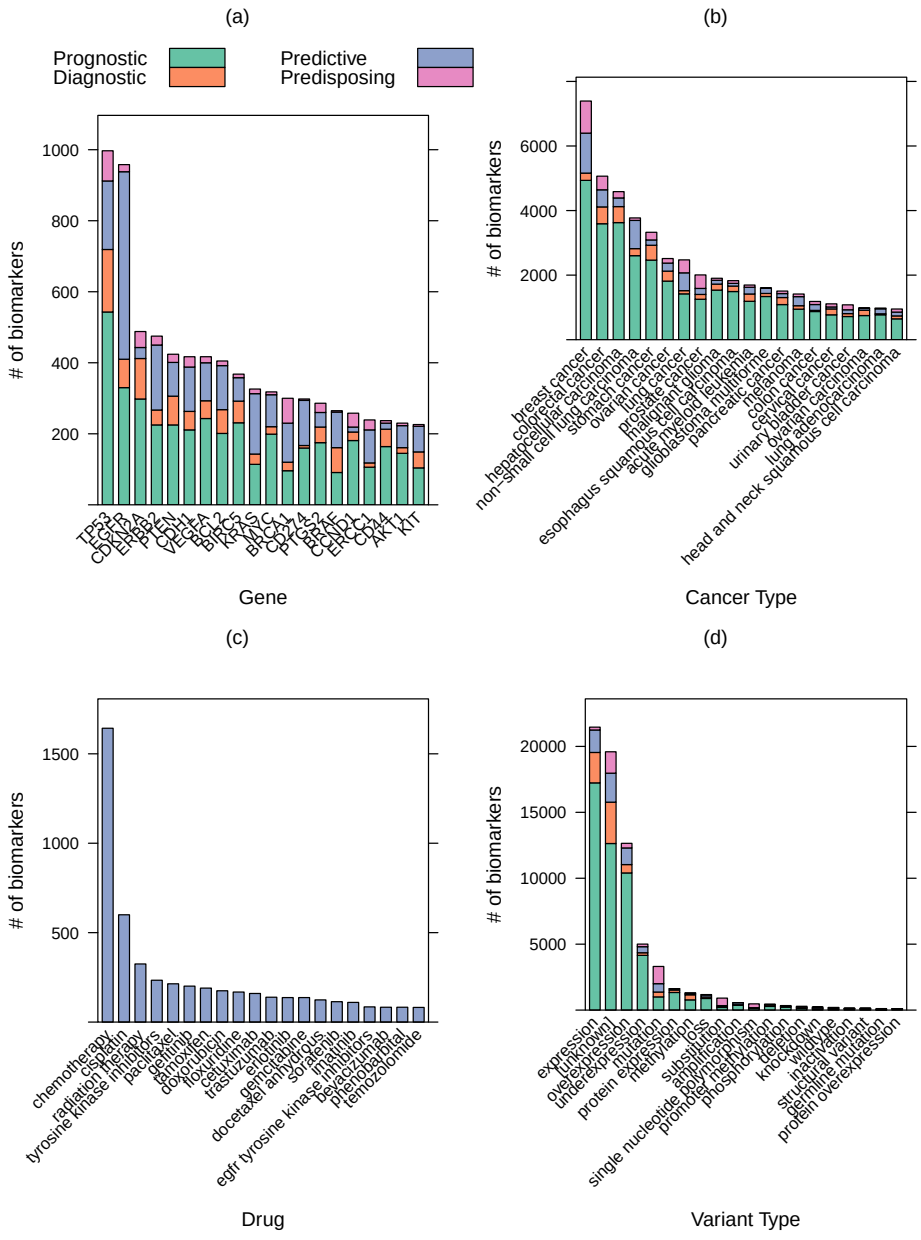


Figure 5.7: An overview of the top 20 (a) genes, (b) cancer types, (c) drugs and (d) variants extracted as part of evidence items.

For the predictive evidence type, we see a disproportionally large number associated with the general term chemotherapy and specific types of chemotherapy including cisplatin, paclitaxel and doxorubicin (Fig 5.7c). Many targeted therapies are also frequently discussed such as the EGFR inhibitors, gefitinib, erlotinib and cetuximab. More general terms such as "tyrosine kinase inhibitor" capture biomarkers related to drug families.

Lastly, we see that expression related biomarkers dominate the variant types (Fig 5.7d). Markers based on expression are more likely to be prognostic than those using non-expression data (81.3% versus 45.6%). The easiest method to explore the importance of a gene in a cancer type is to correlate expression levels with patient survival. With the accessibility of large transcriptome sets and survival data (e.g. TCGA), such assertions have become very common. The 'mutation' variant type has a more even split across the four evidence types. The mutation term covers very general phrasing without a specific mention of the actual mutation. The substitution variant type does capture this information but there are far fewer. This reflects the challenge of extracting all the evidence item information from a single sentence. It is more likely for an author to define a mutation in another sentence and then use a general term (e.g. EGFR mutations) when discussing its clinical relevance. There are also a substantial number of evidence items where the variant cannot be identified and are flagged as '[unknown]'. These are still valuable but may require more in-depth curation in order to tease out the actual variant.

Of all the biomarkers extracted, 21.1% (14,931/ 70,655) are supported by more than one citation. In fact, the most cited biomarker is BRCA1 mutation as a predisposing marker in breast cancer with 545 different papers discussing this. The initial priority for CIViC annotation is on highly cited biomarkers that have not yet been curated into CIViC, in order to eliminate obvious information gaps. However, the single citations may also represent valuable information for precision cancer analysts and CIViC curators focused on specific genes or diseases.

We compared the 70,655 biomarkers extracted for CIViCmine with the 2,055 in the CIViC resource as of 05 June 2018. Figure 5.8a shows the overlap of exact evidence items between the two resources. The overlap is quite small and the number in CIViCmine not included in CIViC is very large. We next compare the cited publications using PubMed ID. Despite not having used CIViC publications in training CIViCmine, we find that a substantial number of papers cited in CIViC (253/1,325) were identified automatically

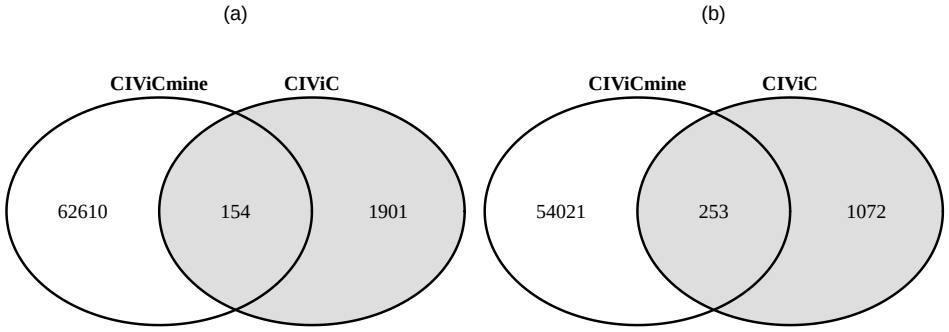


Figure 5.8: A comparison of the evidence items curated in CIViC and automatically extracted by CIViCmine by (a) exact biomarker information and by (b) paper.

by CIViCmine (Fig 5.8b). Altogether, CIViCmine includes 5,568 genes, 388 cancer types and 272 drugs or drug families not yet included in CIViC.

5.3.1 Use Cases

There are two use cases of this resource that are already been realised by CIViC curators at the McDonnell Genome Institute and analysts at the BC Cancer Agency.

Knowledge base curation use case: The main purpose of this tool is to assist in curation of new biomarkers in CIViC. A CIViC curator, looking for a frequently discussed biomarker, would access the CIViCmine Shiny app through a web browser. This would present the table, pie charts and filter options on the left. They would initially filter the CIViCmine results for those not already in CIViC. If they had a particular focus, they may filter by Evidence Type. For example, some CIViC curators may be more interested in Diagnostic, Predictive and Prognostic biomarkers than Predisposing. This is due to the focus on somatic events in many cancer types. They would then look at the table of biomarkers, already sorted by citation count in descending order, and select the top one. This would then populate a table further down the page. Assuming that this is a frequently cited biomarker, there would be many sentences discussing it, which would quickly give the curator a broad view of whether it is accepted in the community. They would then open multiple tabs on their web browser to start

looking at several of the papers discussing it. They might select an older paper, close to when it was first established as a biomarker, and a more recent paper from a high-impact journal to gauge the current view of the biomarker. Several of the sentences may obviously cite other papers as being important to establishing this biomarker. The curator would look at these papers in particular, as they may be the most appropriate to curate. Importantly, the curator may want the primary literature source(s), which includes the experimental data supporting this biomarker.

Personalized cancer analyst use case: While interpreting an individual patient tumor sample, an analyst typically needs to interpret a long list of somatic events. Instead of searching PubMed for each somatic event, they can initially check CIViC and CIViCmine for existing structured knowledge on the clinical relevance of each somatic event. First, they should check CIViC given the high level of pre-existing curation there. This would involve searching the CIViC database through their website or API. If the variant does not appear there, they would then progress to CIViCmine. By using the filters and search functionality, they could quickly narrow down the biomarkers for their gene and cancer type of interest. If a match is found, they can then move to the relevant papers that are listed below to understand the experiments that were done to make this assertion. If they agree with the biomarker, they could then suggest it as a curated biomarker for the CIViC database. Both CIViC and CIViCmine reduce curation burden by aggregating likely applicable data across multiple synonyms for the gene, disease, variant or drug not as easily identified through PubMed searches.

5.4 Discussion

This work provides several significant contributions to the fields of biomedical text mining and precision oncology. Firstly, the annotation method is drastically different from previous approaches. Most annotation projects (such as the BioNLP Shared Tasks (Kim et al., 2009, Kim et al. (2011)) and the CRAFT corpus (Bada et al., 2012)) have focused on abstracts or entire documents. The biomarkers of interest for this project appear sparsely in papers so it would have been inappropriate to annotate full documents and a focus on individual sentences was necessary. We identified sentences that contained the appropriate entities and then filtered them further in order to provide a rich set that contained similar numbers of relevant sentences as irrelevant sentences that could then be annotated. This approach could be

applied to many other biomedical topics.

We also made use of a simpler annotation system than the often used brat (Stenetorp et al., 2012) which allowed for fast annotation by restricting the possible annotation options. Specifically, annotators did not select the entities but were shown all appropriate permutations that matched the possible relation types. Issues of incorrect entity annotation were reported through the interface, collated and used to make improvements to the underlying wordlists for gene, cancer types and drugs. We found that once a curator became familiar with the task, they could curate sentences relatively quickly. Expert annotation is key to providing high quality data to build and evaluate a system. Therefore reducing the time required for expert annotators is essential.

The supervised learning approach differs from methods that used co-occurrence based (STRING) or rule-based (mirTex) methods. Firstly, the method is able to extract complex meaning from the sentence providing results that would be impossible with a co-occurrence method. A rule-based method would require enumerating the possible ways of describing each of the diverse evidence types. Our approach is able to capture a wide variety of biomarker descriptions. Furthermore most relation extraction methods aim for optimal F1-score (Chaix et al., 2016), placing an equal emphasis on precision as recall. With the goal of minimizing false positives, our approach of high precision and low recall would be an appropriate model for other information extraction methods applied to the vast PubMed corpus.

Apart from the advantages outlined previously, several other factors lead to the decision to use a supervised learning approach to build this knowledge base. The CIViC knowledge base could have been used as training data in some form. The papers already in CIViC could have been searched for the sentences discussing the relevant biomarker, which could then have been used to train a supervised relation extraction system. An alternative approach to this problem would have been to use a distant supervision method using the CIViC knowledge base as seed data. This approach was taken by Peng et al who also attempted to extract relations across sentence boundaries (Peng et al., 2017). They chose to focus only on point mutations and extracted 530 within sentence biomarkers and 1,461 cross-sentence biomarkers. These numbers are drastically smaller than the 70,655 extracted in CIViCmine.

The reason to not use the CIViC knowledge base in the creation of the training data was taken to avoid any curator-specific bias that may have

formed in the selection of papers and biomarkers to curate. This was key to providing a broad and unbiased view of the biomarkers discussed in the literature. CIViC evidence items include additional information such as directionality of a relationship (e.g. does a mutation cause drug sensitivity or resistance), the level of support for it (from preclinical models up to FDA guidelines) and several other factors. It is highly unlikely that all this information will be included within a single sentence. Therefore, we did not try to extract this information concurrently. Instead, it is an additional task for the curator as they process the CIViCmine prioritised list.

A robust named entity recognition solution does not exist for a custom term list of cancer types, drugs and variants. For instance, the DNorm tool does not capture many cancer subtypes. A decision was made to go for high recall for entity recognition, including genes, as the relation extraction step would then filter out many incorrect matches based on context. This decision is further supported by the constant evolution of cancer type ontologies as demonstrated by workshops at recent Biocuration conferences.

Finally, this research provides a valuable addition to the precision oncology informatics community. CIViCmine can be used to assist curation of other precision cancer knowledge bases and can be used directly by precision cancer analysts to search for biomarkers of interest. As this resource will be kept up-to-date with the latest research, it will likely constantly change as new cancer types and drug names enter the lexicon. We hope that the methods described can be used in other biomedical domains and that the resources provided will be valuable to the biomedical text mining and precision oncology fields.

Chapter 6

Conclusions

At inception of this thesis work, we hoped that text mining could someday become an everyday tool for the biomedical research community. We were specifically interested in the use of text mining to collate knowledge for the personalized oncology field. This final chapter will discuss how the work undertaken has contributed to these goals and what hurdles remain. We will broadly discuss the lessons learnt during this thesis and suggest interesting future directions to pursue, particularly to overcome some of the limitations acknowledged within this work.

6.1 Contributions

Many research areas are overwhelmed by potential hypotheses to test and automated hypothesis generation methods are designed to provide prioritized lists to researchers. Several factors limit these methods being embraced by the biology research community, including the predictive performance, explainability, and poor awareness that these methods exist. Our work in Chapter 2 pushed forward the predictive performance by developing and evaluating a new approach using co-occurrence data. We showed that our SVD-based method outperformed the previously best performing methods and explored the explainability of some the successful and failed predictions.

Supervised relation extraction is an important step past co-occurrences in information extraction. Our work with the VERSE and Kindred tools in Chapter 3 illustrated that vectorized dependency path-based approaches are the best method for biomedical relation extraction and that deep learning does not yet achieve the same benefits in other fields with larger training dataset sizes. The VERSE system won part of the BioNLP Shared Task 2016. Furthermore, the packaging of Kindred makes it easier for other researchers to use our methods for their own problems.

Our CancerMine resource, described in Chapter 4, will benefit all cancer biology researchers as a valuable tool to understand the role of different genes in cancer. The high-precision knowledge extraction pipeline proves that single sentences do contain enough information for large-scale knowledge base construction. By examining the frequently cited gene roles, we were able to build profiles for each cancer type that can be used to find similarities between cancers and were validated by comparison to data in the Cancer Genome Atlas (TCGA) project.

Finally Chapter 5 describes the CIVICmine resource designed specifically for curating information about the growing field of precision oncology and the clinical relevance of mutations in cancer. This resource will prove increasingly valuable in the coming years as more medical centres develop precision oncology programs. The methods for annotating the training data and building a classifier that can scale to PubMed provide valuable guidelines for other groups interested in building a high-precision knowledge base in another area of biology.

6.2 Lessons Learnt

The stated goal of much biomedical text mining research is to help biologists and medical researchers absorb research and identify potential hypotheses for study. With the information overload present in published literature, automated methods should be used to guide researchers to the knowledge that they need. Throughout this thesis work, I have identified several key problems that frequently occur in biomedical text mining. These problems are fruitful areas for future research.

6.2.1 Inaccessible and out-of-date results

Firstly, and importantly, access to text mined results is key to adoption by researchers. Many research papers develop text mining methods where the code and/or data are not shared. These papers may benefit other text mining researchers with algorithmic improvement ideas or approaches that could be generalized to other text mining problem. But they do not help biologists.

Text mining published literature has been a focus of research for several decades. Advances in computational power within the last 15 years has

made it possible to do large-scale processing of a large number of PubMed abstracts and full-text papers. Hence there have been multiple analyses of PubMed data, but very few are kept up-to-date as new publications are added to the corpus.

The reason for this lack of updating is primarily that researchers move onto other projects after publication and potentially move to other institutions (especially graduate students). The additional engineering required to maintain text mining results can be too much for a research group. But if text mining is to become a ubiquitous tool for biologists, this must be a problem that is overcome and would be a valuable direction for future work.

6.2.2 User Interfaces

The way that a biologist can interact with the text-mined data is key. Even if the data is public, most biologists do not understand the value of text mining and would not go to the effort of downloading data and searching it themselves. Hence a user interface is absolutely essential for this development. To be more specific, a graphical user interface is required as few biologists would be willing to use a command-line application.

There are three common paths for building applications with graphical user interfaces. First, the tool can be implemented as a standalone desktop application. These require installation and are often operating system specific (e.g. only running on Windows). The second is as a Java application that can be launched from a website. More web browsers are blocking Java applications by default due to the high-security risks involved in executing a Java application (e.g. access to full file system).

This brings us to the third option which I would argue is the only real option these days. With advances in web technologies, specifically AJAX-like libraries, that provide responsive websites for high-quality user experiences, web apps are the best solution. These can be client-side only where all calculations and analysis are done using Javascript code. Or more commonly, with a server-side end with a database, text mining results can be queried quickly. Several bioinformatics analysis tools have been frequently due to their implementation as web applications. The DAVID tool for gene set enrichment analysis (Dennis et al., 2003) is a classic example of a tool that is frequently used when other more up-to-date tools exist but are hard to use.

These arguments lead us to build web apps for the CancerMine and CIViCmine projects. We used the Shiny web technology for its ease of implementation and visually attractive interfaces. Unfortunately Shiny may not scale well to a larger number of users and these interfaces may be revisited if the resources prove very popular. We would encourage other text mining developers to consider providing a web interface to navigate text-mined data.

There is a huge area of research in human-computer interaction (HCI). It could easily be argued that there should be more integration between text mining and HCI research in order to understand what features make a tool easier to use. If a biologist finds a tool frustrating to use, or the results unreliable, they may never use the tool again. The CancerMine and CIViCmine research, fortunately, took place in an environment close to potential users of these resources which provided the opportunity to discuss their design. Understanding the real needs of users and the challenges they face interpreting text-mined data would enable text mining to become a more valuable part of the research process.

6.3 Limitations and Future Directions

One of the main limitations of our work is the focus on the knowledge contained within single sentences. For all of our projects, we only capture co-occurrences or relations that are discussed within a sentence and do not capture knowledge that is spread across multiple sentences. This is a common limitation of many text mining tools at the moment due to the challenge presented by anaphora. Coreference resolution methods still provide noisy results when identifying which specific term a pronoun (or general noun) refer to. We examined the ability to extract relations across sentence boundaries but found (as others have) that the false positive rate skyrockets as more sentences are included. This is largely due to the decrease in class balance, as the positive examples become a small fraction of all possible candidate relations. Overcoming this limitation with a high-quality coreference resolution method would provide the largest gain for relation extraction methods used to populate knowledge bases (as in Chapters 4 and 5).

We are also limited by access to text corpora for information extraction. We chose to focus on PubMed and PubMed Central Open Access subset (PMCOA) as they contain the largest set of published abstracts and full-text

articles while also being the easiest to access. Several publishers are beginning to make other smaller corpora accessible through limited APIs (and often requiring special permissions) (Westergaard et al., 2018). However, these new corpora provide additional challenges with unique file formats and rights permissions when sharing the results of text mining. This will be the primary stumbling block of biomedical text mining in the coming decades. Several universities have shown the desire to change their relationships with publishers to encourage easier access to literature, both for text mining and for researchers in general. We hope these efforts progress quickly.

In Chapters 4 and 5, we faced a common problem in biomedical text mining. For supervised learning, annotated training data is needed to build a classifier. The size of the training data is a limiting factor for the complexity of the classifier that can be built. The recent successes of deep learning in other fields, particularly computer vision, have been led by the development of vast training sets (e.g. ImageNet (Deng et al., 2009)). In fact, Google acquired reCAPTCHA in order to generate human annotated image data to improve their computer vision algorithms for Google Streetview and Project Gutenberg (Von Ahn et al., 2008). For the biomedical field, expert annotators may be needed for specific tasks. Some researchers have tried crowdsourcing (e.g. Mark2theCure (Tsueng et al., 2016)) either through volunteers or Mechanical Turk paid workers (Buhrmester et al., 2011). These crowdsourcing efforts have shown that many non-expert annotators must look at the same sentence in order to get a good consensus. This increases the annotation cost and drove our decision to use expert annotators for CancerMine and CIViCmine. However, it created the limitation of a smaller training set size. This smaller training set size meant that a deep learning based approach wasn't a viable approach given the currently established issue with overfitting small data set sizes (Mehryary et al., 2016). The BioNLP Shared Tasks showed that more classical approaches, as taken in Chapter 3, were still the most reliable approach for relation extraction given smaller training set sizes.

An interesting angle that should be pursued is active learning in which the data for annotation is continuously updated to identify the most confusing sentences for the system. This approach is impeded by the need to use multiple annotators and would likely require small batch active learning instead of continually updated active learning.

The decision to focus on a limited set of relations between the biomedical

entities of interest (e.g. genes and cancers) has advantages and disadvantages. In Chapter 4, we were interested in only three relation types (Drivers, Oncogenes and Tumor Suppressors). There are many other relations that can exist between a gene and a cancer type, e.g. "frequently mutated in". By focussing on only three relation types, we could provide a tightly controlled annotation process with a specific annotation manual. This meant that the annotation task was feasible and could be completed by annotators within an acceptable amount of time. However, we may be missing interesting relations between these entities. Other approaches take an Open Information Extraction (OpenIE) approach where no assumptions are made about the types of relations that may exist (Percha et al., 2018). An approach that could bridge the two methods would be a valuable addition to the biomedical text mining field.

6.4 Final Words

Biomedical text mining should be an every-day tool used by researchers to keep up-to-date with research and help guide their hypothesis generation. To get to this stage, we have contributed several key ideas, methods, and data-sets, including high precision relation extraction for knowledge base construction. This is an exciting period for this field with the culmination of affordable computational resources, web technologies and advances in biomedical sciences. We must work closely with biomedical researchers to understand the problems that matter to them and enable them to interrogate the biomedical knowledge in a form suited to them.

Bibliography

Adamson, P. C., Houghton, P. J., Perilongo, G., and Pritchard-Jones, K. (2014). Drug discovery in paediatric oncology: roadblocks to progress. *Nature Reviews Clinical Oncology*, 11(12):732.

Aerts, S., Haeussler, M., Van Vooren, S., Griffith, O. L., Hulpiau, P., Jones, S. J., Montgomery, S. B., and Bergman, C. M. (2008). Text-mining assisted regulatory annotation. *Genome Biology*, 9(2):R31.

Altman, R. B. (2018). Challenges for training translational researchers in the era of ubiquitous data. *Clinical Pharmacology & Therapeutics*, 103(2):171–173.

Ananiadou, S. and Mcnaught, J. (2006). *Text mining for biology and biomedicine*. Artech House London.

Anekalla, K. R., Courneya, J., Fiorini, N., Lever, J., Muchow, M., and Busby, B. (2017). Pubrunner: a light-weight framework for updating text mining results. *F1000Research*, 6.

Athenikos, S. J. and Han, H. (2010). Biomedical question answering: A survey. *Computer methods and programs in biomedicine*, 99(1):1–24.

Bada, M., Eckert, M., Evans, D., Garcia, K., Shipley, K., Sitnikov, D., Baumgartner, W. A., Cohen, K. B., Verspoor, K., Blake, J. A., and others (2012). Concept annotation in the craft corpus. *BMC bioinformatics*, 13(1):161.

Baylin, S. B. and Ohm, J. E. (2006). Epigenetic gene silencing in cancer–a mechanism for early oncogenic pathway addiction? *Nature Reviews Cancer*, 6(2):107.

Bennett, J., Lanning, S., and others (2007). The netflix prize. In *Proceedings of KDD cup and workshop*, volume 2007, page 35. New York, NY, USA.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Björne, J. and Salakoski, T. (2013). TEES 2.1: Automated annotation scheme learning in the BioNLP 2013 Shared Task. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 16–25.

Björne, J. and Salakoski, T. (2015). Tees 2.2: biomedical event extraction for diverse corpora. *BMC bioinformatics*, 16(16):S4.

Bodenreider, O. (2004). The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Bohannon, J. (2016). Who's downloading pirated papers? everyone.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117.

Bruskiewich, R., Huellas-Bruskiewicz, K., Ahmed, F., Kaliyaperumal, R., Thompson, M., Schultes, E., Hettne, K. M., Su, A. I., and Good, B. M. (2016). Knowledge. bio: A web application for exploring, building and sharing webs of biomedical relationships mined from pubmed. *bioRxiv*, page 055525.

Buhrmester, M., Kwang, T., and Gosling, S. D. (2011). Amazon's mechanical turk: A new source of inexpensive, yet high-quality, data? *Perspectives on psychological science*, 6(1):3–5.

Bui, Q.-C., Campos, D., van Mulligen, E., and Kors, J. (2013). A fast rule-based approach for biomedical event extraction. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 104–108. Association for Computational Linguistics.

Bunescu, R. C. and Mooney, R. J. (2005). A shortest path dependency kernel for relation extraction. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 724–731. Association for Computational Linguistics.

Burgstaller-Muehlbacher, S., Waagmeester, A., Mitraka, E., Turner, J., Putman, T., Leong, J., Naik, C., Pavlidis, P., Schriml, L., Good, B. M., and others (2016). Wikidata as a semantic framework for the gene wiki initiative. *Database*, 2016.

Carpenter, T. and Thatcher, S. G. (2014). The challenges of bibliographic control and scholarly integrity in an online world of multiple versions of journal articles. *Against the Grain*, 23(2):5.

Chaix, E., Dubreucq, B., Fatihi, A., Valsamou, D., Bossy, R., Ba, M., Delèger, L., Zweigenbaum, P., Bessieres, P., Lepiniec, L., and others (2016). Overview of the regulatory network of plant seed development (seedev) task at the bionlp shared task 2016. In *Proceedings of the 4th BioNLP Shared Task Workshop*, pages 1–11.

Chakravarty, D., Gao, J., Phillips, S., Kundra, R., Zhang, H., Wang, J., Rudolph, J. E., Yaeger, R., Soumerai, T., Nissan, M. H., and others (2017). Oncokb: a precision oncology knowledge base. *JCO precision oncology*, 1:1–16.

Chang, M. T., Asthana, S., Gao, S. P., Lee, B. H., Chapman, J. S., Kandoth, C., Gao, J., Socci, N. D., Solit, D. B., Olshen, A. B., and others (2016). Identifying recurrent mutations in cancer reveals widespread lineage diversity and mutational specificity. *Nature biotechnology*, 34(2):155.

Cheng, J., Demeulemeester, J., Wedge, D. C., Vollan, H. K. M., Pitt, J. J., Russnes, H. G., Pandey, B. P., Nilsen, G., Nord, S., Bignell, G. R., and others (2017). Pan-cancer analysis of homozygous deletions in primary tumours uncovers rare tumour suppressors. *Nature Communications*, 8(1):1221.

Chu, J., Lauretti, E., Di Meco, A., and Pratico, D. (2013). Flap pharmacological blockade modulates metabolism of endogenous tau in vivo. *Translational psychiatry*, 3(12):e333.

Ciccarelli, F. D., Venkata, S. K., Repana, D., Nulsen, J., Dressler, L., Bortolomeazzi, M., Tourna, A., Yakovleva, A., and Palmieri, T. (2018). The network of cancer genes (ncg): a comprehensive catalogue of known and

candidate cancer genes from cancer sequencing screens. *bioRxiv*, page 389858.

Clark, K. and Manning, C. D. (2015). Entity-centric coreference resolution with model stacking. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1405–1415.

Comeau, D. C., Batista-Navarro, R. T., Dai, H.-J., Doğan, R. I., Yepes, A. J., Khare, R., Lu, Z., Marques, H., Mattingly, C. J., Neves, M., and others (2014). Bioc interoperability track overview. *Database*, 2014.

Comeau, D. C., Doğan, R. I., Ciccarese, P., Cohen, K. B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M., and others (2013). Bioc: a minimalist approach to interoperability for biomedical text processing. *Database*, 2013.

Council, N. R. and others (2014). *Convergence: facilitating transdisciplinary integration of life sciences, physical sciences, engineering, and beyond*. National Academies Press.

Davies, M. (2009). The 385+ million word corpus of contemporary american english (1990–2008+): Design, architecture, and linguistic insights. *International journal of corpus linguistics*, 14(2):159–190.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE.

Dennis, G., Sherman, B. T., Hosack, D. A., Yang, J., Gao, W., Lane, H. C., and Lempicki, R. A. (2003). David: database for annotation, visualization, and integrated discovery. *Genome biology*, 4(9):R60.

Developers, J. (2008). Jython implementation of the high-level, dynamic, object-oriented language python written in 100% pure Java. Technical report, Technical report (1997-2016), http://www. jython. org/(accessed May 2016).

DiGiacomo, R. A., Kremer, J. M., and Shah, D. M. (1989). Fish-oil dietary supplementation in patients with raynaud's phenomenon: a double-blind, controlled, prospective study. *The American journal of medicine*, 86(2):158–164.

Eckart, C. and Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218.

Ernst, P., Meng, C., Siu, A., and Weikum, G. (2014). Knowlife: a knowledge graph for health and life sciences. In *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*, pages 1254–1257. IEEE.

Fiorini, N., Canese, K., Starchenko, G., Kireev, E., Kim, W., Miller, V., Osipov, M., Kholodov, M., Ismagilov, R., Mohan, S., and others (2018). Best match: new relevance search for pubmed. *PLoS biology*, 16(8):e2005343.

Fiorini, N., Lipman, D. J., and Lu, Z. (2017). Cutting edge: Towards pubmed 2.0. *eLife*, 6:e28801.

Forbes, S. A., Beare, D., Gunasekaran, P., Leung, K., Bindal, N., Boutselakis, H., Ding, M., Bamford, S., Cole, C., Ward, S., and others (2014). Cosmic: exploring the world's knowledge of somatic mutations in human cancer. *Nucleic acids research*, 43(D1):D805–D811.

Frijters, R., Heupers, B., van Beek, P., Bouwhuis, M., van Schaik, R., de Vlieg, J., Polman, J., and Alkema, W. (2008). Copub: a literature-based keyword enrichment tool for microarray data analysis. *Nucleic acids research*, 36(suppl_2):W406–W410.

Funk, C., Baumgartner, W., Garcia, B., Roeder, C., Bada, M., Cohen, K. B., Hunter, L. E., and Verspoor, K. (2014). Large-scale biomedical concept recognition: an evaluation of current automatic annotators and their parameters. *BMC bioinformatics*, 15(1):59.

Futreal, P. A., Coin, L., Marshall, M., Down, T., Hubbard, T., Wooster, R., Rahman, N., and Stratton, M. R. (2004). A census of human cancer genes. *Nature Reviews Cancer*, 4(3):177.

Gala, K., Li, Q., Sinha, A., Razavi, P., Dorso, M., Sanchez-Vega, F., Chung, Y. R., Hendrickson, R., Hsieh, J., Berger, M., and others (2018). Kmt2c mediates the estrogen dependence of breast cancer through regulation of erα enhancer function. *Oncogene*.

Gonzalez-Perez, A., Perez-Llamas, C., Deu-Pons, J., Tamborero, D., Schroeder, M. P., Jene-Sanz, A., Santos, A., and Lopez-Bigas, N. (2013). Intogen-mutations identifies cancer drivers across tumor types. *Nature methods*, 10(11):1081.

Good, B. M., Ainscough, B. J., McMichael, J. F., Su, A. I., and Griffith, O. L. (2014). Organizing knowledge to enable personalization of medicine in cancer. *Genome biology*, 15(8):438.

Gordon, M. D. and Dumais, S. (1998). Using latent semantic indexing for literature based discovery. *Journal of the American Society for Information Science*.

Griffith, M., Spies, N. C., Krysiak, K., McMichael, J. F., Coffman, A. C., Danos, A. M., Ainscough, B. J., Ramirez, C. A., Rieke, D. T., Kujan, L., and others (2017). Civic is a community knowledgebase for expert crowdsourcing the clinical interpretation of variants in cancer. *Nature genetics*, 49(2):170.

Haber, D. A. and Settleman, J. (2007). Cancer: drivers and passengers. *Nature*, 446(7132):145.

Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pages 11–15, Pasadena, CA USA.

Hanahan, D. and Weinberg, R. A. (2000). The hallmarks of cancer. *cell*, 100(1):57–70.

Hegi, M. E., Diserens, A.-C., Gorlia, T., Hamou, M.-F., de Tribolet, N., Weller, M., Kros, J. M., Hainfellner, J. A., Mason, W., Mariani, L., and others (2005). Mgmt gene silencing and benefit from temozolomide in glioblastoma. *New England Journal of Medicine*, 352(10):997–1003.

Hersh, W. (2008). Information retrieval in literature-based discovery. In *Literature-based Discovery*, pages 153–172. Springer.

Hettne, K. M., Thompson, M., van Haagen, H. H., Van Der Horst, E., Kaliyaperumal, R., Mina, E., Tatum, Z., Laros, J. F., Van Mulligen, E. M., Schuemie, M., and others (2016). The Implicitome: A resource for rationalizing gene-disease associations. *PloS one*, 11(2):e0149621.

Hirschman, L., Yeh, A., Blaschke, C., and Valencia, A. (2005). Overview of biocreative: critical assessment of information extraction for biology.

Honnibal, M. and Johnson, M. (2015). An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

Hristovski, D., Rindflesch, T., and Peterlin, B. (2013). Using literature-based discovery to identify novel therapeutic approaches. *Cardiovascular & Hematological Agents in Medicinal Chemistry (Formerly Current Medicinal Chemistry-Cardiovascular & Hematological Agents)*, 11(1):14–24.

Huang, B., Qu, Z., Ong, C. W., Tsang, Y. N., Xiao, G., Shapiro, D., Salto-Tellez, M., Ito, K., Ito, Y., and Chen, L.-F. (2012). Runx3 acts as a tumor suppressor in breast cancer by targeting estrogen receptor α. *Oncogene*, 31(4):527.

Huang, L., Fernandes, H., Zia, H., Tavassoli, P., Rennert, H., Pisapia, D., Imielinski, M., Sboner, A., Rubin, M. A., Kluk, M., and others (2017). The cancer precision medicine knowledge base for structured clinical-grade mutations and interpretations. *Journal of the American Medical Informatics Association*, 24(3):513–519.

Jelier, R., Schuemie, M. J., Roes, P.-J., van Mulligen, E. M., and Kors, J. A. (2008a). Literature-based concept profiles for gene annotation: the issue of weighting. *International journal of medical informatics*, 77(5):354–362.

Jelier, R., Schuemie, M. J., Veldhoven, A., Dorssers, L. C., Jenster, G., and Kors, J. A. (2008b). Anni 2.0: a multipurpose text-mining tool for the life sciences. *Genome Biol*, 9(6):R96.

Jones, S. J., Laskin, J., Li, Y. Y., Griffith, O. L., An, J., Bilenky, M., Butterfield, Y. S., Cezard, T., Chuah, E., Corbett, R., and others (2010). Evolution of an adenocarcinoma in response to selection by targeted kinase inhibitors. *Genome biology*, 11(8):R82.

Kang, R., Li, H., Ringgaard, S., Rickers, K., Sun, H., Chen, M., Xie, L., and Bünger, C. (2014). Interference in the endplate nutritional pathway causes intervertebral disc degeneration in an immature porcine model. *International orthopaedics*, 38(5):1011–1017.

Kilicoglu, H., Fiszman, M., Rodriguez, A., Shin, D., Ripple, A., and Rindflesch, T. C. (2008). Semantic medline: a web application for managing the results of pubmed searches. In *Proceedings of the third international symposium for semantic mining in biomedicine*, volume 2008, pages 69–76.

Kim, J.-D., Ohta, T., Pyysalo, S., Kano, Y., and Tsujii, J. (2009). Overview of bionlp'09 shared task on event extraction. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task*, pages 1–9. Association for Computational Linguistics.

Kim, J.-D., Ohta, T., Tateisi, Y., and Tsuj, J. (2003). Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl_1):i180–i182.

Kim, J.-D., Pyysalo, S., Ohta, T., Bossy, R., Nguyen, N., and Tsujii, J. (2011). Overview of bionlp shared task 2011. In *Proceedings of the BioNLP shared task 2011 workshop*, pages 1–6. Association for Computational Linguistics.

Kim, J.-D. and Wang, Y. (2012). Pubannotation: a persistent and sharable corpus and annotation repository. In *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, pages 202–205. Association for Computational Linguistics.

Kristensen, V. N., Lingjærde, O. C., Russnes, H. G., Vollan, H. K. M., Frigessi, A., and Børresen-Dale, A.-L. (2014). Principles and methods of integrative genomic analyses in cancer. *Nature Reviews Cancer*, 14(5):299–313.

Lawrence, S. and Giles, C. L. (2000). Accessibility of information on the web. *intelligence*, 11(1):32–39.

Leaman, R. and Gonzalez, G. (2008). Banner: an executable survey of advances in biomedical named entity recognition. In *Biocomputing 2008*, pages 652–663. World Scientific.

Leaman, R., Islamaj Doğan, R., and Lu, Z. (2013). Dnorm: disease name normalization with pairwise learning to rank. *Bioinformatics*, 29(22):2909–2917.

Leaman, R., Wei, C.-H., and Lu, Z. (2015). tmchem: a high performance approach for chemical named entity recognition and normalization. *Journal of cheminformatics*, 7(1):S3.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

Lever, J. and Jones, S. (2017). Painless relation extraction with kindred. *BioNLP 2017*, pages 176–183.

Li, C., Rao, Z., and Zhang, X. (2016). LitWay, Discriminative Extraction for Different Bio-Events. *Proceedings of the 4th BioNLP Shared Task Workshop*, page 32.

Li, G., Ross, K. E., Arighi, C. N., Peng, Y., Wu, C. H., and Vijay-Shanker, K. (2015). mirtex: a text mining system for mirna-gene relation extraction. *PLoS computational biology*, 11(9):e1004391.

Liben-Nowell, D. and Kleinberg, J. (2007). The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031.

Lichtnwalter, R. and Chawla, N. V. (2012). Link prediction: fair and effective evaluation. In *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, pages 376–383. IEEE Computer Society.

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, pages 2181–2187.

Liu, Y., Sun, J., and Zhao, M. (2017). Ongene: a literature-based database for human oncogenes. *Journal of Genetics and Genomics*, 44(2):119–121.

Low, Y., Gonzalez, J. E., Kyrola, A., Bickson, D., Guestrin, C. E., and Hellerstein, J. (2014). Graphlab: A new framework for parallel machine learning. *arXiv preprint arXiv:1408.2041*.

Lu, Z. (2011). Pubmed and beyond: a survey of web tools for searching biomedical literature. *Database*, 2011.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Mehryary, F., Björne, J., Pyysalo, S., Salakoski, T., and Ginter, F. (2016). Deep Learning with Minimal Training Data: TurkuNLP Entry in the

BioNLP Shared Task 2016. *Proceedings of the 4th BioNLP Shared Task Workshop*, page 73.

Mendel, G. and Tschermak, E. (1866). Versuche über pflanzen-hybriden.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Nédellec, C., Bossy, R., Kim, J.-D., Kim, J.-J., Ohta, T., Pyysalo, S., and Zweigenbaum, P. (2013). Overview of bionlp shared task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 1–7.

Nickel, M., Tresp, V., and Kriegel, H.-P. (2012). Factorizing yago: scalable machine learning for linked data. In *Proceedings of the 21st international conference on World Wide Web*, pages 271–280. ACM.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., and others (2016). Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666.

Onitilo, A. A., Engel, J. M., Greenlee, R. T., and Mukesh, B. N. (2009). Breast cancer subtypes based on er/pr and her2 expression: comparison of clinicopathologic features and survival. *Clinical medicine & research*, 7(1-2):4–13.

Pan, R., Zhou, Y., Cao, B., Liu, N. N., Lukose, R., Scholz, M., and Yang, Q. (2008). One-class collaborative filtering. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 502–511. IEEE.

Panyam, N. C., Khirbat, G., Verspoor, K., Cohn, T., and Ramamohanarao, K. (2016). SeeDev Binary Event Extraction using SVMs and a Rich Feature Set. *Proceedings of the 4th BioNLP Shared Task Workshop*, page 82.

Patterson, S. E., Liu, R., Statz, C. M., Durkin, D., Lakshminarayana, A., and Mockus, S. M. (2016). The clinical trial landscape in oncology and connectivity of somatic mutational profiles to targeted therapies. *Human genomics*, 10(1):4.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., and others (2011a). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011b). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Peng, N., Poon, H., Quirk, C., Toutanova, K., and tau Yih, W. (2017). Cross-sentence n-ary relation extraction with graph lstms. *Transactions of the Association for Computational Linguistics*, 5:101–115.

Percha, B., Altman, R. B., and Wren, J. (2018). A global network of biomedical relationships derived from text. *Bioinformatics*, 1:11.

Prasad, V., Fojo, T., and Brada, M. (2016). Precision oncology: origins, optimism, and potential. *The Lancet Oncology*, 17(2):e81–e86.

Prud'hommeaux, E. and Seaborne, A. (2006). SPARQL query language for RDF. Technical report.

Quinn, C. T., Johnson, V. L., Kim, H.-Y., Trachtenberg, F., Vogiatzi, M. G., Kwiatkowski, J. L., Neufeld, E. J., Fung, E., Oliveri, N., Kirby, M., and others (2011). Renal dysfunction in patients with thalassaemia. *British journal of haematology*, 153(1):111–117.

Radtke, F. and Raj, K. (2003). The role of notch in tumorigenesis: oncogene or tumour suppressor? *Nature Reviews Cancer*, 3(10):756.

Ramakrishnan, C., Patnia, A., Hovy, E., and Burns, G. A. (2012). Layout-aware text extraction from full-text pdf of scientific articles. *Source code for biology and medicine*, 7(1):7.

RStudio, Inc (2013). *Easy web applications in R.* URL: http://www.rstudio.com/shiny/.

Rüdiger, T., Ott, G., Ott, M. M., Müller-Deubert, S. M., and Müller-Hermelink, H. K. (1998). Differential diagnosis between classic hodgkin's lymphoma, t-cell-rich b-cell lymphoma, and paragranuloma by paraffin immunohistochemistry. *The American journal of surgical pathology*, 22(10):1184–1191.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088):533.

Schriml, L. M., Arze, C., Nadendla, S., Chang, Y.-W. W., Mazaitis, M., Felix, V., Feng, G., and Kibbe, W. A. (2011). Disease ontology: a backbone for disease semantic integration. *Nucleic acids research*, 40(D1):D940–D946.

Schult, D. A. and Swart, P. (2008). Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conferences (SciPy 2008)*, volume 2008, pages 11–16.

Shimomura, O., Johnson, F. H., and Saiga, Y. (1962). Extraction, purification and properties of aequorin, a bioluminescent protein from the luminous hydromedusan, aequorea. *Journal of Cellular Physiology*, 59(3):223–239.

Shrager, J. and Tenenbaum, J. M. (2014). Rapid learning for precision oncology. *Nature Reviews Clinical oncology*, 11(2):109–118.

Smith, T. (2015). How far can biology's big data take us? Vancouver Bioinformatics User Group (VanBUG) Seminar Series.

Smith, T. F. and Waterman, M. S. (1980). New stratigraphic correlation techniques. *The Journal of Geology*, 88(4):451–457.

Smith, T. F. and Waterman, M. S. (1981). Comparison of biosequences. *Advances in applied mathematics*, 2(4):482–489.

Soon, W. M., Ng, H. T., and Lim, D. C. Y. (2001). A machine learning approach to coreference resolution of noun phrases. *Computational linguistics*, 27(4):521–544.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.

Swanson, D. R. (1986a). Fish oil, raynaud's syndrome, and undiscovered public knowledge. *Perspectives in biology and medicine*, 30(1):7–18.

Swanson, D. R. (1986b). Undiscovered public knowledge. *The Library Quarterly*, 56(2):103–118.

Swanson, D. R. and Smalheiser, N. R. (1997). An interactive system for finding complementary literatures: a stimulus to scientific discovery. *Artificial intelligence*, 91(2):183–203.

Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K. P., and others (2014). String v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic acids research*, 43(D1):D447–D452.

Szklarczyk, D., Morris, J. H., Cook, H., Kuhn, M., Wyder, S., Simonovic, M., Santos, A., Doncheva, N. T., Roth, A., Bork, P., and others (2016). The string database in 2017: quality-controlled protein–protein association networks, made broadly accessible. *Nucleic acids research*, page gkw937.

Tamborero, D., Rubio-Perez, C., Deu-Pons, J., Schroeder, M. P., Vivancos, A., Rovira, A., Tusquets, I., Albanell, J., Rodon, J., Tabernero, J., and others (2018). Cancer genome interpreter annotates the biological and clinical relevance of tumor alterations. *Genome medicine*, 10(1):25.

Taschuk, M. and Wilson, G. (2017). Ten Simple Rules for Making Research Software More Robust. *PLOS Computational Biology*, 13(4).

Tsatsaronis, G., Balikas, G., Malakasiotis, P., Partalas, I., Zschunke, M., Alvers, M. R., Weissenborn, D., Krithara, A., Petridis, S., Polychronopoulos, D., and others (2015). An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138.

Tsueng, G., Nanis, S., Fouquier, J., Good, B., and Su, A. (2016). Citizen science for mining the biomedical literature. *Citizen Science: Theory and Practice*, 1(2).

Tsuruoka, Y., Miwa, M., Hamamoto, K., Tsujii, J., and Ananiadou, S. (2011). Discovering and visualizing indirect associations between biomedical concepts. *Bioinformatics*, 27(13):i111–i119.

Tsuruoka, Y., Tateishi, Y., Kim, J.-D., Ohta, T., McNaught, J., Ananiadou, S., and Tsuj, J. (2005). Developing a robust part-of-speech tagger for biomedical text. In *Advances in informatics*, pages 382–392. Springer.

Van Landeghem, S., Björne, J., Wei, C.-H., Hakala, K., Pyysalo, S., Ananiadou, S., Kao, H.-Y., Lu, Z., Salakoski, T., Van de Peer, Y., and others

(2013). Large-scale event extraction from literature with multi-level gene normalization. *PloS one*, 8(4):e55814.

Von Ahn, L., Maurer, B., McMillen, C., Abraham, D., and Blum, M. (2008). recaptcha: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468.

Vrandečić, D. and Krötzsch, M. (2014). Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Wagner, A. H., Walsh, B., Mayfield, G., Tamborero, D., Sonkin, D., Krysiak, K., Pons, J. D., Duren, R., Gao, J., McMurry, J., and others (2018). A harmonized meta-knowledgebase of clinical interpretations of cancer genomic variants. *bioRxiv*, page 366856.

Wei, C.-H., Harris, B. R., Kao, H.-Y., and Lu, Z. (2013a). tmVar: a text mining approach for extracting sequence variants in biomedical literature. *Bioinformatics*.

Wei, C.-H., Kao, H.-Y., and Lu, Z. (2013b). Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, 41(W1):W518–W522.

Weinstein, J. N., Collisson, E. A., Mills, G. B., Shaw, K. R. M., Ozenberger, B. A., Ellrott, K., Shmulevich, I., Sander, C., Stuart, J. M., Network, C. G. A. R., and others (2013). The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113.

Westergaard, D., Stærfeldt, H.-H., Tønsberg, C., Jensen, L. J., and Brunak, S. (2018). A comprehensive and quantitative comparison of text-mining in 15 million full-text articles versus their corresponding abstracts. *PLoS computational biology*, 14(2):e1005962.

Weymann, D., Laskin, J., Roscoe, R., Schrader, K. A., Chia, S., Yip, S., Cheung, W. Y., Gelmon, K. A., Karsan, A., Renouf, D. J., and others (2017). The cost and cost trajectory of whole-genome analysis guiding treatment of patients with advanced cancers. *Molecular genetics & genomic medicine*, 5(3):251–260.

Wiles, A. (1995). Modular elliptic curves and fermat's last theorem. *Annals of mathematics*, 141(3):443–551.

William, H. (2007). Numerical recipes: The art of scientific computing. 3rd edition.

Yetisgen-Yildiz, M. and Pratt, W. (2009). A new evaluation methodology for literature-based discovery systems. *Journal of biomedical informatics*, 42(4):633–643.

Zender, L., Xue, W., Zuber, J., Semighini, C. P., Krasnitz, A., Ma, B., Zender, P., Kubicka, S., Luk, J. M., Schirmacher, P., and others (2008). An oncogenomics-based in vivo rnai screen identifies tumor suppressors in liver cancer. *Cell*, 135(5):852–864.

Zhao, M., Kim, P., Mitra, R., Zhao, J., and Zhao, Z. (2015). Tsgene 2.0: an updated literature-based knowledgebase for tumor suppressor genes. *Nucleic acids research*, 44(D1):D1023–D1031.

Zhu, S., Zeng, J., and Mamitsuka, H. (2009). Enhancing medline document clustering by incorporating mesh semantic similarity. *Bioinformatics*, 25(15):1944–1951.