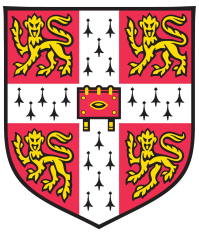
Semantic text classification for cancer text mining



University of Cambridge

Simon Baker

Trinity Hall

September 2017

This dissertation is submitted for the degree of
Doctor of Philosophy

Abstract

Cancer researchers and oncologists benefit greatly from text mining major knowledge sources in biomedicine such as PubMed. Fundamentally, text mining depends on accurate text classification. In conventional natural language processing (NLP), this requires experts to annotate scientific text, which is costly and time consuming, resulting in small labelled datasets. This leads to extensive feature engineering and handcrafting in order to fully utilise small labelled datasets, which is again time consuming, and not portable between tasks and domains.

In this work, we explore emerging neural network methods to reduce the burden of feature engineering while outperforming the accuracy of conventional pipeline NLP techniques. We focus specifically on the cancer domain in terms of applications, where we introduce two NLP classification tasks and datasets: the first task is that of semantic text classification according to the Hallmarks of Cancer (HoC), which enables text mining of scientific literature assisted by a taxonomy that explains the processes by which cancer starts and spreads in the body. The second task is that of the exposure routes of chemicals into the body that may lead to exposure to carcinogens.

We present several novel contributions. We introduce two new semantic classification tasks (the hallmarks, and exposure routes) at both sentence and document levels along with accompanying datasets, and implement and investigate a conventional pipeline NLP classification approach for both tasks, performing both intrinsic and extrinsic evaluation. We propose a new approach to classification using multi-level embeddings and apply this approach to several tasks; we subsequently apply deep learning methods to the task of hallmark classification and evaluate its outcome. Utilising our text classification methods, we develop and two novel text mining tools targeting real-world cancer researchers. The first tool is a cancer hallmark text mining tool that identifies association between a search query and cancer hallmarks; the second tool is a new literature-based discovery (LBD) system designed for the cancer domain. We evaluate both tools with end users (cancer researchers) and find they demonstrate good accuracy and promising potential for cancer research.

Declaration

I declare that this dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface (Acknowledgements) and specified in the text.

It is not substantially the same as any that I have submitted, or, is being concurrently submitted for a degree or diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. I further state that no substantial part of my dissertation has already been submitted, or, is being concurrently submitted for any such degree, diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text

This dissertation contains fewer than 60,000 words excluding appendices, bibliography and figures, but including footnotes and tables, and has fewer than 150 figures.

Simon Baker,
September 2017

"One having bulging tumours.
An ailment with which I will contend."

The Edwin Smith Papyrus—xv 14-15

A surgeon's commentary taken from the oldest known description of cancer in recorded literature.

Acknowledgements

First and foremost, I am immensely grateful to my supervisor Dr. Anna Korhonen, who has guided me through this challenging and rewarding journey. I am especially grateful for her kindness and generosity, and her willingness to leave no stone unturned during the numerous occasions when I needed help.

I am deeply thankful to my examiners Dr. Nigel Collier and Dr. Mark Stevenson for their thorough assessment and interesting discussion of the content presented in this dissertation.

I am grateful to Dr. Sampo Pyysalo who has helped with various technical challenges during my last two years, and I especially thank him for his help in building the CHAT user interface, and his collaboration on the LION project.

I am grateful to Dr. Yufan Guo, who helped me take my first steps in BioNLP during my first year.

I thank Stefan Haselwimmer and Tejas Shah for their help in developing the user interface for LION LBD.

I thank our collaborating cancer researchers from Karolinska Institutet, Sweden, for providing their invaluable expertise and time, without their help much of this work would not be possible. By name, I thank Dr. Imran Ali, Dr. Ilona Silins, Dr. Kristin Larsson, Dr. Johan Högberg, Dr. Ulla Stenius, and Dr. Marika Berglund.

In the same spirit, I thank our expert researchers from CRUK Cambridge Institute, Narita group. I thank Dr. Masashi Narita, and Dr. Shamith Samarajiwa.

I thank Yiannos Stathopoulos for his help and advice with using Lucene.

I am very thankful to the Commonwealth Scholarship Commission and the Cambridge Trust for funding my first three years of my PhD. I also thank Google for funding my final year via the Google Faculty Award through my supervisor.

I thank the Computer Laboratory and Trinity Hall for providing me with travel grants and scholarships during my PhD program. I also thank Lise Gough for her kind help and assistance over the years.

I am forever grateful to my dear parents, who have made untold sacrifices and taken unimaginable risk in order to provide me and my two siblings with better lives and opportunities than what they had. I hope that I will make you proud.

Last but not least, I am especially grateful to Tyler Griffiths, who has volunteered endless hours to proofread and edit this dissertation, and kept me sane during my panic attacks. I will never forget your help and kindness.

Contents

List of Tables 9

List of Figures 11

List of Equations 13

List of Abbreviations 15

Notation 17

1 Introduction 18
- 1.1 Dissertation outline . . . 22
- 1.2 Relevant publications . . . 23

2 Background 25
- 2.1 Biomedical natural language processing . . . 25
  - 2.1.1 Basic tools and resources . . . 25
  - 2.1.2 Information extraction . . . 27
  - 2.1.3 Applications . . . 30
- 2.2 Text mining for cancer . . . 31
- 2.3 Established machine learning algorithms . . . 32
  - 2.3.1 Support vector machines . . . 33
  - 2.3.2 The continuous bag of words model . . . 34
  - 2.3.3 The skip-gram model . . . 35
- 2.4 Standard evaluation metrics . . . 37
- 2.5 Chapter summary . . . 39

3 Semantic text classification for cancer research 41
- 3.1 Related work . . . 41
- 3.2 The Hallmarks of Cancer . . . 43
  - 3.2.1 Annotated corpus . . . 46
  - 3.2.2 Methodology . . . 49
  - 3.2.3 Intrinsic evaluation . . . 52
  - 3.2.4 Case studies . . . 55

3.2.5 Discussion . . . 60
3.3 The Exposure Taxonomy . . . 64
3.3.1 Annotated corpus . . . 65
3.3.2 Methodology . . . 68
3.3.3 Intrinsic evaluation . . . 68
3.3.4 Case studies . . . 69
3.3.5 Discussion . . . 76
3.4 Chapter Summary . . . 77

4 Neural methods 78
4.1 Multi-level embeddings . . . 78
4.1.1 Related work . . . 79
4.1.2 Methodology . . . 79
4.1.3 Evaluation . . . 83
4.1.4 Results . . . 85
4.1.5 Discussion . . . 87
4.2 Convolutional neural networks . . . 88
4.2.1 Related work . . . 89
4.2.2 Methodology . . . 89
4.2.3 Evaluation . . . 91
4.2.4 Results . . . 92
4.2.5 Discussion . . . 95
4.3 Hierarchical text classification . . . 96
4.3.1 Related work . . . 97
4.3.2 Methodology . . . 98
4.3.3 Evaluation . . . 100
4.3.4 Results . . . 102
4.3.5 Discussion . . . 104
4.4 Chapter Summary . . . 105

5 Real-world applications 107
5.1 Cancer hallmark analytics tool . . . 107
5.1.1 Functionality overview . . . 108
5.1.2 Implementation . . . 110
5.1.3 Evaluation . . . 113
5.1.4 Discussion . . . 118
5.2 The LION system for literature-based discovery . . . 120
5.2.1 The ABC model . . . 121
5.2.2 Functionality overview . . . 122
5.2.3 Co-occurrence metrics . . . 128
5.2.4 Implementation . . . 130
5.2.5 Evaluation . . . 134
5.2.6 Discussion . . . 138
5.3 Chapter summary . . . 140

6 Conclusions **141**
6.1 Key contributions 141
6.2 Future directions 144

Bibliography **147**

Appendix A Supplementaries **164**

Appendix B Software and resources **170**
B.1 External software and resources 170
B.2 Developed software and resources 171

List of Tables

2.1 Summary of common resources in BioNLP 26
2.2 Examples of key biomedical corpora and resources for NER 28
2.3 Examples of biomedical resources for relation and event extraction . . 29
2.4 Examples of text mining systems for cancer 32

3.1 Hallmarks and their search terms. 47
3.2 Examples of hallmark annotated sentences 47
3.3 HoC corpus summary statistics 48
3.4 HoC intrinsic evaluation results 53
3.5 HoC leave-one-out feature analysis 54
3.6 HoC case study 3 search queries 58
3.7 HoC case study 3 results 59
3.8 HoC case study 4 search queries 59
3.9 HoC case study 4 results 60
3.10 Hallmark co-occurrence distribution in the annotated corpus. 63
3.11 Hallmark co-occurrence distribution as predicted by the classifier . . . 63
3.12 The Exposure taxonomy 66
3.13 The Exposure taxonomy corpus summary statistics 67
3.14 Exposure taxonomy intrinsic evaluation results 70
3.15 Exposure taxonomy leave-one-out feature analysis 71
3.16 Exposure taxonomy case study 3 search queries 75
3.17 Exposure taxonomy case study 3 results 76

4.1 Description and distribution of argumentative zones 84
4.2 HoC multi-level embeddings experiment results 85
4.3 Document and sentence classification F_1-score 86
4.4 Rhetorical text multi-level embeddings results 87
4.5 CNN model parameters 90
4.6 Word embedding initialisation for MLE 90
4.7 Annotation statistics 91
4.8 CNN performance with oversampling 93
4.9 CNN performance (F_1-score) results 95
4.11 Baseline CNN hierarchical multi-label model 99

4.12 Hierarchical classification—summary data statistics 100
4.13 Jaccard similarity between label pairs 100
4.14 Performance results for Tasks 1 and 2 102
4.15 Hierarchical classification—exact matches results 103
4.16 Post-processing label correction for transitive and retractive correction 104

5.1 CHAT classifiers intrinsic evaluation results 114
5.2 CHAT intrinsic evaluation results 115
5.3 CHAT leave-one-out feature analysis 116
5.4 Examples of active LBD systems 123
5.5 Spearman's rank correlation of LION co-occurrence metrics 135
5.6 Time slicing LBD evaluation 136
5.7 Closed discovery evaluation cases 137
5.8 Closed discovery evaluation results 138

A.1 Hallmarks of cancer taxonomy description 164
A.2 HoC classifiers features summary statistics 167
A.3 Exposure taxonomy classifiers features summary statistics 168
A.4 CHAT classifiers features summary statistics 169

List of Figures

1.1 Growth of publications containing the term “cancer” 19

2.1 CBOW model 35
2.2 The skip-gram model 36

3.1 The Hallmarks of Cancer taxonomy 45
3.2 Annotated labels per example in the HoC corpus 49
3.3 Chapter 3 NLP pipeline 50
3.4 HoC case study 1 distribution profiles 56
3.5 HoC case study 2 distribution profiles 57
3.6 Distribution of correctly predicted hallmarks 61
3.7 The process of chemical risk assessment 64
3.8 Annotated labels per example in the Exposure taxonomy corpus 68
3.9 Classifier performance on the Exposure taxonomy corpus 69
3.10 Exposure taxonomy case study 1 distribution profiles 72
3.11 Exposure taxonomy case study 2 (exposure routes and biomarkers) profiles 74
3.12 Exposure taxonomy case study 2 (effect biomarkers) profiles 74

4.1 Illustration of distributed joint learning 81
4.2 CNN network architecture 89
4.3 CNN performance with different embeddings 93
4.4 CNN performance (AUC) results 94
4.5 Hierarchical multi-label classification 97
4.6 Hierarchical multi-label classification initialisation 98
4.7 Co-occurance initialisation schemes 99
4.8 Post-processing label correction 101
4.9 Hierarchical classification—labels per instance results 103

5.1 CHAT results screen 109
5.2 CHAT abstract view screen 109
5.3 CHAT comparison screen 110
5.4 CHAT NLP pipeline 111
5.5 CHAT user case studies results 119
5.6 Closed discovery (LBD) 122

5.7 Open discovery (LBD) 122
5.8 LION LBD system operation and basic use case 124
5.9 Query to concept disambiguation screen in LION 125
5.10 LION LBD closed discovery concept graph 125
5.11 LION LBD open discovery concept graph 126
5.12 Displaying mention-level information in LION 126
5.13 LION search results in text mode 127
5.14 Ilustration of LION LBD concept graph and co-occurrence 128
5.15 Construction pipeline for LION LBD system 132

List of Equations

2.1 SVM functional margin constraint . . . 33
2.2 SVM optimisation objective . . . 33
2.3 SVM objective function dual form . . . 33
2.4 SVM linear kernel . . . 33
2.5 SVM RBF kernel . . . 34
2.6 CBOW hidden layer output . . . 35
2.7 CBOW output SoftMax function . . . 35
2.8 The CBOW loss function . . . 35
2.9 Skip-gram hidden-layer output . . . 36
2.10 Skip-gram output SoftMax function . . . 36
2.11 The skip-gram loss function . . . 36
2.12 Standard accuracy . . . 37
2.13 Standard precision . . . 37
2.14 Standard recall . . . 38
2.15 F_1-score . . . 38
2.16 Macro-average of standard metrics . . . 38
2.17 Micro-average of standard metrics . . . 38
2.18 Area under ROC curve . . . 39
2.19 Estimation of AUC . . . 39
4.1 Multilevel embedding objective function . . . 80
4.2 Multilevel embedding output softmax formulation . . . 80
4.3 Word distance (WORD-DIST) feature . . . 82
4.4 Earth mover's distance . . . 82
4.5 EMD constraints . . . 82
4.6 Logistic (sigmoid) activation function . . . 98
4.7 Binary cross-entropy loss function . . . 98
4.8 Upper bound of normalised initialisation . . . 99
4.9 Jaccard similarity . . . 100
5.1 Conditional probability of hallmarks given query . . . 108
5.2 PMI between hallmarks and query . . . 108
5.3 NPMI between hallmarks and query . . . 108
5.4 PMI for co-occurrences in LION LBD . . . 128
5.5 NPMI for co-occurrences in LION LBD . . . 128
5.6 Symmetric conditional probability . . . 129

5.7 Jaccard similarity 129
5.8 χ^2 co-occurrence statistic 129
5.9 Student's t-test co-occurrence statistic 129
5.10 Log-likelihood ratio 129
5.11 Log-likelihood ratio approximation with LION parameters 130

List of Abbreviations

4-NP 4-Nonylphenol
ANN Artificial Neural Network
AUC Area Under Receiver Operating Characteristic Curve
AZ Argumentative Zoning
BBzP Butylbenzylphthalate
BI-RADS Breast Imaging Reporting and Data System
BioNLP Biomedical Natural Language Processing
BOW Bag of Words
CBOW Continuous Bag of Words
CHAT Cancer Hallmark Analytics Tool
CHEM Chemicals List (feature type)
CNN Convolutional Neural Network
CRA Cancer Risk Assessment
CRF Conditional Random Field
CSIS Cancer Stage Interpretation System
DBP Dibutylphthalate
DEHP Di(2-ethylhexyl)phthalate
DiBP Diisobutylphthalate
DiDP Diisodecylphthalate
DiNP Diisononylphthalate
DM Distributed Memory
EGF Epidermal Growth Factor
EMD Earth Mover's Distance
GAN Generative Adversarial Network
GAPDH Glyceraldehyde 3-phosphatase dehydrogenase
GR Grammatical Relation
HBM Human Biomonitoring
HCB Hexachlorobenzene
HKG Housekeeping gene
HoC Hallmarks of Cancer
IE Information Extraction

List of Abbreviations

IR Information Retrieval
LBD Literature-Based Discovery
LBOW Lemmatised Bag of Words
LLR Log Likelihood Ratio
LSTM Long Short-Term Memory
MeSH Medical Subject Heading
ML Machine Learning
MLE Multilevel Embedding
MLP Multilayer Perceptron
NE Named Entity
NER Named Entity Recognition
NLP Natural Language Processing
NPMI Normalised Point-wise Mutual Information
OOV Out of Vocabulary
OVR One-*vs*.-Rest
PMI Point-wise Mutual Information
POS Part of Speech
RBF Radial Basis Function
ReLU Rectified Linear Unit
ROC Receiver Operating Characteristic
SCP Symmetric Conditional Probability
SD Semantic Distance (feature type)
SGNS Skip-Gram with Negative Sampling
SNOMED CT Systematized Nomenclature of Medical – Clinical Terms
SVM Support Vector Machine
TBP TATA-Box Binding Protein
TM Text Mining
UI User Interface
VAE Variational Autoencoder
VC Verb Class
VEGF Vascular Endothelial Growth Factor
WSD Word-sense Disambiguation

Notation

Types

$\vec{u}$	Vector: lowercase, with arrow.
$\vec{u}_i$	ith element of vector.
$\mathbf{M}$	Matrix: uppercase, bold.
$\mathbf{M}_{ij}$	ijth element of matrix.
T	Tensor: sans-serif, bold.
$\mathsf{T}_{ijk\ldots}$	$ijk\ldots$th element of tensor.
S	Set: uppercase.

Common operations

$\lvert\vec{u}\rvert$	L_1 norm of vector $\vec{u}$.
$\lVert\vec{u}\rVert$	L_2 norm of vector $\vec{u}$.
$\vec{u}^\frown\vec{v}$	Concatenation of vectors $\vec{u}$ and $\vec{v}$.
$\vec{u}\cdot\vec{v}$	Dot-product of vectors $\vec{u}$ and $\vec{v}$.
$\vec{u}\times\vec{v}$	Cross-product of vectors $\vec{u}$ and $\vec{v}$.
$\mathbf{M}^\top$	Transpose of $\mathbf{M}$.

Miscellaneous

MODEL-NAME	A particular model or baseline.
`resource-name`	A tool, resource, or corpus.
www.example.com	A web address or URL.

Chapter 1

Introduction

According to the latest World Cancer Report (McGuire, 2016), the number of newly diagnosed cases of cancer worldwide in 2012 is estimated at 14.1 million. In the same year, there were 8.3 million cancer related deaths. More than one in three people will develop some form of cancer during their lifetime.

Much progress has been made in cancer research during the past century. Currently, half of people diagnosed with cancer are expected to survive for at least five years after diagnosis. In some countries, this figure is even higher—half the people diagnosed with cancer in the UK, for example, are expected to survive beyond ten years: an improvement of 100% over the past 40 years (Cancer Research UK, 2017). However, cancer is still the most common cause of death (29%), and cancer research therefore forms a major component of the world's scientific output. Figure 1.1 shows the exponential growth of publications in `PubMed`[1] that contain the term "cancer" in the title or abstract.

It is difficult (if not impossible) for an individual to continually keep up to date with this growth. In the past twenty years, text mining (TM) and natural language processing (NLP) methods have increased in popularity among researchers in order to capitalise on the wealth of scientific literature (Holzinger *et al.*, 2014). Biomedical natural language processing (BioNLP) and TM applications for cancer have focused on complex interactions in biological systems such as gene regulatory networks, which are often explored to provide insight into the cell proliferation observed in cancer.

A fundamental component of a modern TM system is accurate classifiers. This often requires using supervised machine learning (ML) techniques that learn from example instances of text labelled accurately with *targets*.[2] Typically, domain experts (for example, cancer researchers or medical practitioners) are needed to annotate the text with the target labels, but acquiring qualified experts can be a challenge and the process of data annotation can be prohibitively time consuming. This

[1] https://www.ncbi.nlm.nih.gov/pubmed/

[2] For instance, to train a classifier to predict whether an email is spam, it must be provided many example emails labelled with the target *spam* or *not-spam*, accordingly.

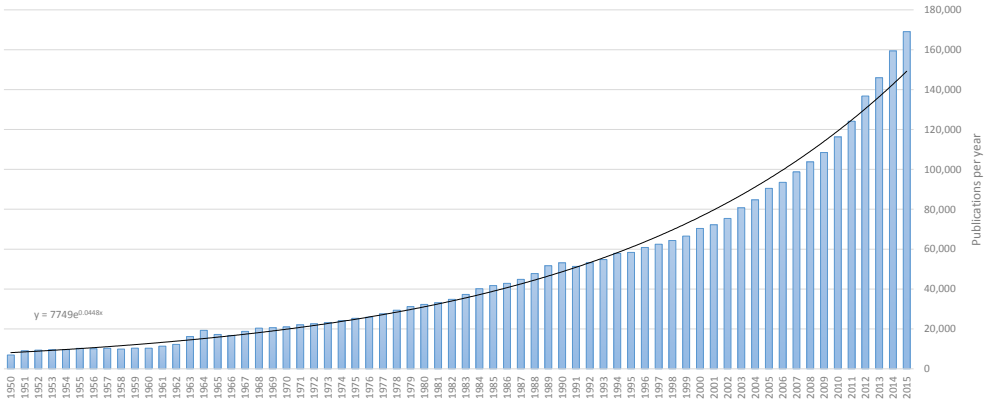


Figure 1.1: The number[3] of newly indexed `PubMed` publications per year that contain "cancer" in the either the title or the abstract. The trendline shows exponential growth since 1950.

typically results in *sparse* datasets—too few examples per label for an ML algorithm to effectively learn to classify text without overfitting to the small set of examples it was trained on.

A common approach to mitigating this problem is to engineer features[4] that can better capture correlations between the raw input text and the target labels in order to better utilise the relatively small amounts of expert annotated data. The feature engineering process as well as extraction of these features from text have several drawbacks. It can be time consuming to handcraft these features, and it can also be computationally demanding to extract these features from text. It may require other dependencies in addition to the raw text input, such as part of speech (POS) tagging, parsing, named entity recognition (NER), *etc.*, which is typically done using an NLP pipeline that extracts these features. Domain experts may still be required to design these features and the methods of acquiring them—for example, when using or constructing domain-specific ontologies. It is therefore a key objective of this work to construct supervised classifiers that can be trained with small and sparse datasets.

Biomedical research, and particularly cancer, is an increasingly multidisciplinary topic (Krallinger *et al.*, 2005). This means that cancer literature can span multiple subdomains, such as clinical research, genomics, chemical risk assessment, or epidemiology. This raises two key challenges:

1. There are many different types of experts (oncologists, genaeologists, epidemiologists, pharmacologists, *etc.*) who may have different areas (domains) of interest in cancer, and would be interested in extracting different sorts of in-

[3] Obtained from https://www.ncbi.nlm.nih.gov/pubmed/?term=cancer

[4] A *feature* is a property or characteristic observed in the raw data input into a classifier. To use the previous spam classification example, a representation of the text of the email may be input as a feature to the classifier.

formation from cancer literature more pertinent to their own areas of interest. Each of these areas can yield additional useful information that should not be missed, so it is important to be able to extract (*i.e.* classify) text from literature encompassing a range of domains, appealing to a wide range of experts.

2. To be practical, BioNLP should be able to cope with domain variation; the task of classification is no exception and classifiers should be able to cope with domain variation in the literature. For example, if an NLP system is trained to identify biological cancer processes in text, then it should be able to handle text from different subdomains such as genomics, or the clinical domain (*e.g.* clinical reports). This is challenging because established ML classifiers require different set of features for different domains; an algorithm trained to classify according to one set of features in one domain will perform poorly in another (Lippincott *et al.*, 2011).

With respect to the first challenge, one objective of this work is to introduce a classification task that benefits a variety of experts that work in most subdomains of cancer. We do so by leveraging a widely-accepted set of principles that explain the inception and proliferation of cancer. As for the second, we investigate several methods that reduce the reliance on heavy feature engineering (*e.g.* automatic feature learning), that would mitigate the problem of training classifiers that work under different subdomains.

Another important issue is the method of evaluation. Although raw scores from intrinsic evaluation provide a good measure to the performance of the classifiers under controlled settings, they cannot give fully insightful information about the utility of the task in the real-world. The predictions of the classifiers might prove to be accurate when evaluated intrinsically on small datasets, but in practice the predictions might not be useful for the intended target users (domain experts). This could be because the NLP task does not solve an interesting real-world problem, or because the methodology is not scalable to real-world data. Although it has not been widely conducted in the BioNLP research community, it is evident that user-centric evaluation is needed in addition to intrinsic methods.

As such, an important objective of this work is evaluating our methodology *via* real-world applications in addition to intrinsic evaluation of the ML classifiers and techniques. This is achieved by engagement of end-users through the construction of tools, as well as numerous expert-driven case studies.

In this work, we make several contributions that address the issues discussed previously. We first introduce two new semantic text classification tasks relevant to cancer. The first concerns the biology of cancer: specifically, the traits that define how cancer forms and proliferates in the body. The second task relates to cancer risk assessment (CRA), and the ways humans are exposed to hazardous chemicals and carcinogens. In collaboration with end-users, we develop annotated corpora and an NLP methodology for classifying text for both of these tasks, and evaluate our approach using both intrinsic and user-centric evaluations using expert-driven case

studies. We then focus on reducing the need for feature engineering by developing neural methods for text classification, exploring a number of these methods with several classification tasks. Finally, we apply these techniques to the building of two tools that are intended for use by real-world cancer researchers.

1.1 Dissertation outline

The remainder of this dissertation is divided into five chapters. The following provides a brief overview of each chapter:

Chapter 2 **Background**
We present background context and related literature in BIONLP and TM for cancer, and also for the basic ML algorithms and evaluation metrics that are used throughout this dissertation.

Chapter 3 **Semantic text classification for cancer research**
We introduce two semantic text classification tasks: the Hallmarks of Cancer (HoC) taxonomy and the Exposure taxonomy. We present an expert-annotated corpus for each task, and we further introduce a feature-rich supervised NLP pipeline for classifying text according to the two taxonomies. We intrinsically evaluate the accuracy of our method and conduct several case studies that show predictions made by our classifiers are confirmed by established verified facts; we obtain levels of accuracy high enough to be suitable for practical tasks in cancer research.

Chapter 4 **Neural methods**
We describe and evaluate novel methods that aim to mitigate the burden of heavy feature-engineering. We first present and evaluate an approach that uses features derived from multi-level embeddings. We then describe convolutional neural networks (CNNs) and apply it to the HoC task. We then describe an approach that for using neural models such as CNNs for hierarchical text classification.

Chapter 5 **Real-world applications**
In this chapter we apply methods described in the previous two chapters to real-world applications by developing two tools for supporting cancer researchers in their work. The first tool is cancer hallmark analytics tool (CHAT), which allows cancer researchers to find correlations between any search term of interest and the HoC. The second tool is a cancer-focused literature-based discovery (LBD) tool that can aid researchers to discover new knowledge from literature by finding previously unmentioned relations between concepts of interest.

Chapter 6 **Conclusions**
We conclude the key findings and contributions of this work and discuss future research directions.

1.2 Relevant publications

In carrying out the work presented in this dissertation, a number of peer reviewed publications were produced. These are briefly described below, and a note is made of the chapters to which they are relevant:

1. **Automatic semantic classification of scientific literature according to the hallmarks of cancer (Baker *et al.*, 2016c).**

   We present the first corpus and NLP pipeline for semantically classifying text according to the HoC. We introduce a corpus of annotated `PubMed` abstracts and train classifiers for this task. We report good performance for both intrinsic and extrinsic evaluations, demonstrating both the accuracy of the methodology and its potential for supporting practical cancer research. *This publication is relevant to Chapter 3.*

2. **Text mining for improved exposure assessment (Larsson *et al.*, 2017).**

   We develop classifiers for chemical exposure information along with a corpus of 3700 annotated abstracts, and a taxonomy created to capture exposure information. Our evaluation demonstrates good performance intrinsically, as well as an improvement in information retrieval for chemical exposure data compared to keyword-based `PubMed` searches. We use a number of case studies to demonstrate that the classifier can be used to assist researchers by facilitating information retrieval and classification. *This publication is relevant to Chapter 3.*

3. **Robust Text Classification for Sparsely Labelled Data Using Multi-level Embeddings (Baker *et al.*, 2016a).**

   We develop a method to jointly learn embeddings at different levels of granularity (word, sentence, and document) along with the class labels. The intuition is that topic semantics represented by embeddings at multiple levels result in better classification. We evaluate this approach in unsupervised and semisupervised settings on two sparsely labelled classification tasks, outperforming the handcrafted models and several embedding baselines. *This publication is relevant to Chapter 4.*

4. **Cancer Hallmark Text Classification Using Convolutional Neural Networks (Baker *et al.*, 2016b).**

   We present a CNN model for the classification of text according to the HoC. The evaluation shows that a basic CNN model can achieve a level of performance competitive with a support vector machine (SVM) trained using complex manually engineered features optimised to the task. We further show that simple modifications to the CNN hyperparameters, initialisation, and training process allow the model to notably outperform the SVM. *This publication is relevant to Chapter 4.*

5. **Initializing neural networks for hierarchical multi-label text classification (Baker and Korhonen, 2017).**

   We present a method for hierarchical multilabel text classification that initialises a neural network model's final hidden layer in such a way that it leverages label co-occurrence relations such as hypernymy. This approach elegantly lends itself to hierarchical classification. We evaluate this approach using two hierarchical multilabel text classification tasks in the biomedical domain using both sentence- and document-level classification. Our evaluation shows promising results. *This publication is relevant to Chapter 4.*

6. **Cancer Hallmarks Analytics Tool (CHAT): A text mining approach to organise and evaluate scientific literature on cancer (Baker *et al.*, 2017).**

   We create an extensive HoC taxonomy and develop an automatic text mining methodology and a tool (CHAT) capable of retrieving millions of cancer-related references from `PubMed` and organising them into the taxonomy. The efficiency and accuracy of the tool was evaluated intrinsically as well as extrinsically by case studies. The correlations identified by the tool show that it offers a great potential to organise and correctly classify cancer-related literature. *This publication is relevant to Chapter 5.*

7. **Improving Literature-Based Discovery with Advanced Text Mining (Korhonen *et al.*, 2015).**[5]

   We discuss how advanced TM based on IR, NLP, and data mining could open the doors to much deeper, wider coverage and dynamic LBD better capable of evolving with science, in particular when combined with sophisticated, state-of-the-art knowledge discovery techniques. *This publication is relevant to Chapter 5.*

8. **LION LBD: a Literature-Based Discovery System for Cancer Biology (Pyysalo *et al.*, 2017).**[6]

   We introduce LION LBD, an LBD system that enables researchers to navigate published information and supports hypothesis generation and testing. The system is built with a particular focus on the molecular biology of cancer. The system is evaluated using four case studies based on landmark discoveries in cancer research. The evaluation demonstrates the system's ability to identify undiscovered links and rank relevant concepts highly among potential connections. *This publication is relevant to Chapter 5.*

[5] The author's contribution to this publication was limited to public presentation of its content.
[6] This publication is under peer review at the time of writing this dissertation.

Chapter 2

Background

This chapter provides the reader with background information relevant to BioNLP and text mining. We also summarise the well-established ML algorithms and evaluation metrics that are used throughout this thesis.

2.1 Biomedical natural language processing

This section describes some basic resources and tools used in BioNLP. The specific task of information extraction (IE) is discussed, followed by an overview of some major applications of IE in BioNLP.

2.1.1 Basic tools and resources

One of the main resources used in BioNLP research is scientific literature, often obtained as either publication abstracts or full publications. The most widely used scientific literature system in BioNLP is `PubMed`, an IR system (search engine) that searches several major databases such as `MEDLINE`. In addition to `PubMed`, the BioNLP community makes use of many ontologies, vocabularies, taxonomies, corpora, and other resources. Some examples of these are given in Table 2.1.

The nature of the language used in biomedical literature differs significantly from general literature. It is often dense with long sentences containing complex clausal structures, using very domain-specific terminology and acronyms (Cohen and Demner-Fushman, 2014). This makes biomedical language particularly challenging for NLP.

The inherent domain-specific linguistic distributions found in biomedical language makes it difficult to use models such as part of speech (POS) taggers and parsers that are trained in the general domain. To solve this issue, tools need to be trained with biomedical domain data for optimal performance. This would require resources such as lexicons, terminologies, and annotated datasets specific to the domain. Biomedical literature is known to have subdomains that have their own distinct linguistic distributions (Lippincott *et al.*, 2011). Consequently, a parser

Table 2.1: Examples of commonly used resources in the BIONLP community.

Resource	Description
PMC (Roberts, 2001)	PubMed Central (PMC) is a free full-text archive of biomedical and life sciences journal literature maintained by the US national library of medicine (NLM). https://www.ncbi.nlm.nih.gov/pmc/
ChEBI (Degtyarenko *et al.*, 2007)	Chemical Entities of Biological Interest (ChEBI) is a dictionary of molecular entities focused on 'small' chemical compounds. https://www.ebi.ac.uk/chebi/
MEDLINE (Greenhalgh, 1997)	MEDLINE is the main bibliographic database of life sciences and biomedical publication queried by PubMed. Maintained by the US NLM
MeSH (Lipscomb, 2000)	MeSH is a widely used vocabulary for tagging and indexing scientific articles; maintained by the US NLM. https://www.nlm.nih.gov/mesh/
NCBI gene (Maglott *et al.*, 2005)	NCBI Previously known as Entrez Gene, NCBI gene is a database of nomenclatures, reference sequences (RefSeqs), maps, pathways, variations, phenotypes *etc.* for multiple species. https://www.ncbi.nlm.nih.gov/gene
NCBI taxonomy (Federhen, 2011)	NCBI taxonomy is a taxonomy database with curated classification and nomenclature for all of the organisms in the public sequence databases, currently containing approximately 10% of known species. https://www.ncbi.nlm.nih.gov/taxonomy
OBO Foundry (Smith *et al.*, 2007)	OBO Foundry is a multi-disciplinary initiative to develop interoperable and well-formed ontologies, which include the Gene Ontology GO. http://www.obofoundry.org/
GO (Ashburner *et al.*, 2000)	Gene Ontology (GO) is an initiative to unify the representation of gene and gene product attributes across all species. It is widely used in BIONLP. http://www.geneontology.org/
PRO (Natale *et al.*, 2010)	Protein Ontology (PRO) is an ontology of protein-related entities, including isoforms and protein complexes. http://proconsortium.org/pro/
UniProt (Consortium *et al.*, 2014)	UniProt is a database of protein sequences and biological functional information derived from the research literature. http://www.uniprot.org/

trained with annotations from molecular biology will not perform optimally when applied to text in another subdomain (*e.g.* psychiatry).

The last decade has seen some progress towards adapting mainstream NLP models and techniques for the biomedical domain. The `C&C`[7] Parser, a lexicalised combinatory catogorical grammar parser (Clark *et al.*, 2002), achieved 81.4% F_1-score accuracy when adapted for the molecular biology domain using the `GENIA corpus`[8] (Rimell and Clark, 2009). Parsers have been developed to address the issue of subdomain variation. For example, the `Charniak/Johnson Penn-Treebank` Parser (McClosky and Charniak, 2008) was self-trained using unlabelled biomedical abstracts, and achieved 84.3% F_1-score with the `GENIA corpus`. In spite of the recent efforts to adapt current technology and resources for the biomedical domain, there still exists a need to develop methods that can be used to create resources like lexicons automatically. It is also important to reduce reliance on annotations in order to build technologies less susceptible to domain variation (Govindarajan and Ravichandran, 2016).

2.1.2 Information extraction

IE in the biomedical domain often refers to the process of extracting biomedical named entities, relations, and events. Named entity recognition (NER) is the task of locating and classifying words and phrases in natural text into predefined categories. In BIONLP, this often includes categories such as proteins, genes, gene mutations, cell types, cell lines, diseases, drugs, organisms, *etc.*

NER for biomedical text differs from general domain text in several important ways:

1. There are many synonyms and alternate spellings for an entity, causing large word vocabulary sizes and reducing the efficiency of gazetteers and dictionaries that are often used in conventional NER tools (Goulart *et al.*, 2011).

2. Biomedical named entities often consist of long sequences of tokens, making harder to detect the boundaries (Leser and Hakenberg, 2005).

3. It is common to refer to entities by abbreviations, which are often not standardised—for example, using polysemy in referring to proteins and genes (Krauthammer and Nenadic, 2004).

The current state-of-the-art NER systems are feature-rich supervised classifiers, with significant domain-specific feature engineering (Tang *et al.*, 2014). The most competitive systems use conditional random fields (CRFs) (Lafferty *et al.*, 2001; Settles, 2004), which typically use features such as character-level *n*-grams, part of speech (POS) tags, and Bag of Words (BOW) features. Prominent NER tools in

[7] http://groups.inf.ed.ac.uk/ccg/software.html
[8] http://www.nactem.ac.uk/aNT/genia.html

Table 2.2: Examples of key biomedical corpora and resources for NER

Corpus	Description
`Acromine` (Okazaki and Ananiadou, 2006)	An abbreviation dictionary that is automatically constructed from `MEDLINE`. http://www.nactem.ac.uk/software/acromine
`BioLexicon` (Thompson *et al.*, 2011a)	A corpus that is annotated by using terms from `UniProtKb`, `ChEBI`, and `NCBI`. http://www.ebi.ac.uk/Rebholz-srv/BioLexicon/biolexicon.html
`GENETAG` (Tanabe *et al.*, 2005)	A corpus of more than 20.000 `MEDLINE` sentences for gene/protein term identification. https://github.com/openbiocorpora/genetag
`NLPBA` (Kim *et al.*, 2004)	A corpus annotated with technical terms from molecular biology. http://www.nactem.ac.uk/tsujii/GENIA/ERtask/report.html
`GENIA` (Kim *et al.*, 2003)	An NER corpus annotated with terms from the `GENIA` ontology covering species, chemicals, cell types, genes/proteins. http://www.nactem.ac.uk/aNT/genia.html

BIONLP that use CRFs include `BANNER`[9] (Leaman *et al.*, 2008) and `ABNER`[10] (Settles, 2005). Many of these tools are trained on corpora described in Table 2.2.

In recent years, biomedical NER tools have utilised artificial neural networks (ANNs). Some of the earliest published work using some elements of neural networks in biomedical NER is by Tang *et al.* (2014), who evaluated the use of distributed representation (embeddings) learned by neural networks on JNLPBA (Kim *et al.*, 2004) and BioCreative NER shared tasks (Rinaldi *et al.*, 2008). More recently, work by Habibi *et al.* (2017) shows that using the combination of a long short-term memory (LSTM) model with a CRF achieves state-of-the-art results for several NER tasks. Crichton *et al.* (2017) have also demonstrated promising results with multitask learning in NER.

In addition to NER, two other active areas of active research in IE are relation extraction and event extraction.

Relation extraction aims to infer connections occurring in text between relevant entities of interest: for instance, protein–protein interactions (Blaschke *et al.*, 1999), drug–drug interactions (Segura-Bedmar *et al.*, 2010), or drug–disease treatment pairs (Xu and Wang, 2013). Table 2.3 lists example mainstream resources used to train some of these systems.

[9] http://banner.sourceforge.net/
[10] http://pages.cs.wisc.edu/bsettles/abner/

Table 2.3: Examples of biomedical resources for relation and event extraction tasks.

Resource	Description
`BioInfer` (Pyysalo *et al.*, 2007)	A corpus for protein–protein interaction. The data of `BioInfer` are combined from other well-known protein–protein interaction corpora: `AIMed`, `LLL`, `IEPA`, and `HPRD50`. http://mars.cs.utu.fi/BioInfer/
`GENIA Meta-knowledge` (Thompson *et al.*, 2011b)	The GENIA Meta-knowledge corpus contains the original GENIA events corpus that consists of 1000 annotated MEDLINE abstracts, which describe events such as gene expression and regulation. http://www.nactem.ac.uk/meta-knowledge/
`PICorpus` (Johnson *et al.*, 2007)	`PICorpus` is a protein–protein interaction corpus used in several IE tasks such as NER, and relation extraction. http://bionlp-corpora.sourceforge.net/picorpus/
`BioCreative III` (Arighi *et al.*, 2011)	Contains corpora for three three tasks: cross-species gene identification and normalization, protein–protein interactions extraction, and interactive gene indexing and retrieval. http://www.biocreative.org/resources/corpora/biocreative-iii-corpus/
`STRING` (Jensen *et al.*, 2008)	A corpus for predicted protein interactions, including physical and functional associations from genomic context. http://www.bork.embl.de/Docu/STRING-IE/

Traditionally, relation extraction in BIONLP has been achieved with rule-based systems and parse trees (Fundel *et al.*, 2006). Current systems typically use supervised classifiers such as SVMs and achieve the state-of-the-art results for several shared tasks (Wei *et al.*, 2016). Recent work utilising sequence neural models have been showing much promise. For example, recent work by Li *et al.* (2017) employed a bidirectional long short-term memory (BiLSTM) model that uses character, word, and parse trees as features for jointly extracting entities and relations, achieving state-of-the-art results on two tasks.

Event extraction in BIONLP typically refers to the automatic detection of biomolecular interactions. Events can capture complex interactions, for example: "Protein A causes protein B to bind protein C". Event extraction systems would extract the semantics of this event from unstructured text, and represent it in a formalised statement (*e.g.* "*cause*(A, *bind*(B, C))"). Such formal structures can be processed with computational methods. The GENIA event extraction shared tasks are examples that aim to extract events involving genes and proteins (Kim *et al.*, 2009, 2011, 2013). Other resources used for to train event extraction system are also listed in Table 2.3.

Successful event extraction systems have typically been heavy on parsing and

domain specific feature engineering (Hogenboom *et al.*, 2016). However, recent systems have proposed new approaches that are less dependent on parsing and feature engineering. For example, work by Natarajan *et al.* (2017) utilises Markov logic networks, and work by (Rao *et al.*, 2017) uses LSTMs.

The performance of IE systems varies significantly between tasks. NER is generally an easier task than relation and event extraction. State-of-the-art NER systems achieve F_1-scores ranging from 75% up to 90% and above for some tasks (Leaman and Lu, 2016), compared to a range of 50% to 85% for relation extraction tasks (Zhou *et al.*, 2014); event extraction tasks have ranged from 20% up to 80% F_1-scores (Rao *et al.*, 2017).

In general, the best performing systems use fully supervised ML algorithms that have been trained with in-domain data relevant to the target tasks. This is not always possible because in-domain annotated data can be difficult to acquire, and this remains on of the key challenges in biomedical IE (Huang and Lu, 2015).

2.1.3 Applications

Here we briefly summarise several relevant, well-known downstream applications that utilise BIONLP.

Pathway curation: the creation and formalisation of biological knowledge has conventionally been achieved through manual expert effort to curate based on evidence found in scientific literature. It is often very challenging for the curators to keep updated with the latest literature. BIONLP applications have been developed in the last decade to assist human curators to speed up the curation process for various curation tasks. Pathway curation is an example of one curation task aimed at extracting information from biomedical scientific literature to produce formal representations of complex biological processes (Ohta *et al.*, 2013).

Ontology extraction: constructing ontologies manually is extremely labour intensive and time consuming. There is a need for utilising NLP techniques to automate this process. Ontology extraction is the automatic or semiautomatic process of learning ontologies from unstructured text. This usually involves extracting domain specific concepts and the relationships and encoding them with an ontology-specific language for storage and retrieval purposes. A recent example of ontology extraction is the BioCreative V (track 4) shared task, which attempts to complete a biological network *via* Biological Expression Language by extracting evidence from natural text and other resources (Rinaldi *et al.*, 2016). Many of the discussed BIONLP and IE techniques are utilised in ontology extraction (Liu *et al.*, 2011).

Information retrieval: the objective of IR systems is to scour unstructured materials such as text documents and filtering and organising relevant information that must be optimised for fast access (retrieval) given a query. Biomedical IR is often challenging for similar reasons as other tasks and applications in BIONLP—for example, new terms are created frequently, and a lack of standardisation leads to inconsistent typographical or lexical variants of the same entities. It is often challenging to offer a unified method for preprocessing, indexing, and retrieving

biomedical information (Hersh, 2008). NLP techniques are often employed in IR systems to handle these challenges (Jacobs, 2014; Strzalkowski, 1999).

Text mining: often used as an umbrella term for many related applications; these include: literature-based discovery—systems that aim to discover new connections between concepts by aggregating large quantities of text (Holzinger *et al.*, 2013), hypothesis generation—systems that suggest novel associations between entities (Petric *et al.*, 2014), and literature networks—graphs for navigating literature around a set of concepts, where the edges are typically hyperlinked to the underlying literature (Liu and Rastegar-Mojarad, 2016).

2.2 Text mining for cancer

A widely used definition of TM (Fleuren and Alkema, 2015) is given by Hearst (1999), a pioneering figure in the field: "The discovery by computer of new, previously unknown information, by automatically extracting and relating information from different written resources, to reveal otherwise 'hidden' meanings". TM has been applied to many areas of biomedicine (Cohen and Hersh, 2005; Fleuren and Alkema, 2015; Zweigenbaum *et al.*, 2007), and cancer research is no exception.

There are more than 100 types of cancer having different causes, symptoms, and treatments. Therefore, cancer covers a diverse set of diseases, which are differentiated by an ever increasing set of biomarkers and other characteristics (Spasić *et al.*, 2014). It is by its nature highly interdisciplinary, and therefore there are many subdomains in literature, such as clinical, genomics, CRA, *etc.* All tend to have different lexical features, and discuss different types of topics. This can make it a difficult task for BIONLP and TM (Spasić *et al.*, 2014; Zhu *et al.*, 2013).

TM systems are traditionally linear in nature, and usually consist of four high-level stages (Zhu *et al.*, 2013). The following summarises these phases for a system that performs full-scale TM leading to knowledge discovery.

Stage 1: Information retrieval and preprocessing
The first phase is obtaining the relevant text from database sources (such as `PubMed`). Some systems are batch based, which are updated infrequently; others are live, querying the source on demand.

Stage 2: Concept recognition
NER tools are used to extract concept mentions of interest—typical examples include gene, protein, mRNA (messenger RNA), miRNA (micro-RNA), metabolism related terms, and cell terms (Jin, 2007).

Stage 3: Relation extraction
Concepts are then in some way connected *via* relations—either labelled or unlabelled relations (*e.g.* co-occurrence based relations) (Li *et al.*, 2008). These can be represented in a complex graph (Nickel *et al.*, 2016), or a relational database (Krallinger *et al.*, 2005). Examples of such relations in cancer TM systems include signalling pathways (Poon *et al.*, 2014), gene–gene interaction and gene expression in cancer (Natarajan *et al.*, 2006), and microRNAs in gene–gene interaction (Xie *et al.*, 2013).

Table 2.4: Example of current text mining systems for cancer.

TM System	Description
`miRCancer` Xie *et al.* (2013)	Rule-based microRNA extraction from `PubMed` text with cancer association. http://mircancer.ecu.edu
`PubMeth` Ongenaert *et al.* (2007)	An annotated cancer DNA methylation database that includes genes that are methylated in various cancer types. http://www.pubmeth.org/
`MeInfoText` Fang *et al.* (2008)	An IR and rule-based TM system for finding associative information between gene methylation and cancer types. http://mit.lifescience.ntu.edu.tw
`CRAB` Guo *et al.* (2014)	A TM tool for chemical cancer risk assessment. Uses supervised ML for the semantic classification of `PubMed` literature according to evidence of carcinogenic activity and modes of action. https://sites.google.com/site/crabtextmining
`PESCADOR` Barbosa-Silva *et al.* (2011)	A web tool that extracts a network of interactions between concepts from `UniProt` from a set of input `PubMed` abstracts. The relation extraction is co-occurrence based only. https://pescador.uni.lu
`LiverCancerMarkerRIF` Dai *et al.* (2014)	A curation tool aided by TM for liver cancer biomarkers narrations. Allows users to curate supporting evidence on liver cancer biomarkers directly while browsing `PubMed`. http://btm.tmu.edu.tw/LiverCancerMarkerRIF

Stage 4: Knowledge discovery
The aim is to find answers for queries using the constructed system. This can be achieved in several ways—for example, exploring a network, hypothesis generation, finding an association, or IR.

Over the past two decades, there have been several TM systems that specifically target the cancer domain. These systems are varied in their applications and subdomains. Some target gene methylation and its association with types of cancer, while others target cancer risk. Table 2.4 lists a number of example cancer TM systems that are currently in active usage.

2.3 Established machine learning algorithms

In this section, we briefly describe several relevant machine learning (ML) algorithms that are prevalent in current NLP and BIONLP, and are extensively used and referred to throughout this work as either a baseline or as a component of an NLP pipeline.

2.3.1 Support vector machines

The SVM algorithm was first introduced by Vapnik (1963), and later extended to include the "kernel trick" (Boser *et al.*, 1992) and the soft-margin (Cortes and Vapnik, 1995).

Fundamentally, the goal of an SVM is to find a hyperplane in a high dimensional space that can be used for classification. The goal is to find a good separation by the hyperplane by keeping the largest distance to the nearest training datapoint (usually called a functional margin). The intuition is that a larger functional margin results in a lower generalisation error for the classifier. Support vector machines are generalised linear classifiers[11] (an extension of the perceptron). They simultaneously minimise the classification error and maximise the geometric margin (known as maximum margin classifiers).

The objective of the SVM algorithm is to maximise $\frac{2}{|\vec{w}|}$ subject to:

$$y\left(\vec{w}\cdot\vec{x}-b\right)\geq 1 \tag{2.1}$$

where $\vec{x}$ is data, y is its label, $\vec{w}$ is a normal vector to the hyperplane, and $\frac{2}{|\vec{w}|}$ is the margin. This is expressed as an optimisation problem in terms of Lagrange multipliers α as follows:

$$\arg\min_{\vec{w},b}\max_{\alpha\geq 0}\left\{\frac{1}{2}|\vec{w}|^2-\sum_{i=1}^{n}\alpha_i\left[y_i\left(\vec{w}\cdot\vec{x}_i-b\right)\right]\right\} \tag{2.2}$$

In dual form, the above optimisation is expressed as the following objective function:

$$\mathcal{L}(\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j k\left(\vec{x_i},\vec{x_j}\right) \tag{2.3}$$

for $i = 1..n$ and $\alpha_i \geq 0$ with the constraint to minimise b: $\sum_{i=1}^{n}\alpha_i\vec{x}_i = 0$, where the term $k\left(\vec{x}_i,\vec{x}_j\right)$ is the kernel function. The kernel function takes the low dimensional feature input space and transforms it to a higher dimensional space, thereby converting a nonseparable problem to a much more separable one. This kernel function must be continuous and symmetric. The most commonly used kernels are linear and the radial basis function (RBF). The linear kernel is the simplest function. It is given by the inner product $\langle x, y\rangle$ plus an optional constant c :

$$K_{\text{linear}}\left(\vec{x},\vec{y}\right)=\vec{x}^{\top}\cdot\vec{y}+c \tag{2.4}$$

[11] A *linear classifier* uses only linear combination of the input features to learn the decision boundary.

where $\vec{x}$ is the input feature vector, $\vec{y}$ is the output label feature vector, and c is a hyperparamater.

The space between the points to be discriminated is not always linearly separable; the input points are mapped into a much higher dimensional space, thereby making the separation easier in that space. The RBF kernel is a nonlinear kernel which can separate the distance between points with a nonlinear hyperplane. It is typically expressed as follows:

$$K_{\text{RBF}}\left(\vec{x}, \vec{y}\right) = \exp\left(\frac{-\left|\vec{x} - \vec{y}\right|^2}{2\sigma^2}\right) \tag{2.5}$$

where σ is a hyperparameter. The performance of an SVM depends on the selection of kernel and its hyperparameters, which is typically achieved using parameter searching on a held-out development dataset.

2.3.2 The continuous bag of words model

Continuous bag of words (CBOW) was introduced by Mikolov *et al.* (2013a,b), and has since attained prominence and widespread usage in the NLP community as part of the `word2vec`[12] tool.

CBOW is an unsupervised representation learning algorithm that functions much like a language model—that is, it has the objective of estimating the probability for tokens (usually words) given the context, *i.e.* $P(w_t \mid c\,(w_t)); \forall w \in V$, where w_t is the target word, V is the vocabulary of the training data, and $c(w_t)$ is the set of context words that surround the target word. The algorithm is designed to learn a vector representation[13] while achieving this objective.

CBOW is an ANN that is composed of an input layer, a fully connected (dense) hidden layer, and an output layer as shown in Figure 2.1.

The input layer size is equal to the vocabulary size of the training data, words[14] are represented as one-hot vectors (a vector of size $|V|$ where one element of the vector is set to 1 to indicate the target word, while the remaining words are set to zero). The hidden layer size is set to the dimensionality of the resulting word vectors, *i.e.*, a hyperparameter to the model, and all of the neurons in the hidden layer are linear units. The size of the output layer is equal to the input layer. Therefore, if the vocabulary for learning word vectors consists of V words and D is the size of the dimension of these word vectors, then the hidden layer connections can be represented by matrix $\mathbf{H}$ of size $V \times D$, where each row represents a word. The hidden layer output is essentially the product of the hidden layer weight matrix (which are the learned embeddings) and the average of the input context vectors $\vec{w}$

[12] https://code.google.com/archive/p/word2vec/

[13] The term *embedding* is typically used to describe a distributed semantic representation, usually expressed as a vector in a vector space model.

[14] For optimisation, `word2vec` uses Huffman codes instead of words.

(denoted in Figure 2.1 by the 'Σ' symbol). We use $\vec{h}$ to denote this vector, which is calculated as follows:

$$\vec{h} = \frac{1}{|c(w_t)|}\left(\sum_{c_i \in c(w_t)} \vec{c}_i\right)^\top \tag{2.6}$$

The final layer generates a probability value for the target word; this is done by converting the activation values output by the hidden layer into probabilities using the SoftMax[15] function as follows:

$$P(w_t \mid c(w_t)) = \frac{\exp\left(\vec{h}^\top \cdot \vec{w}_t\right)}{\sum_{v_i \in V} \exp\left(\vec{h}^\top \cdot \vec{v}_i\right)} \tag{2.7}$$

The loss function L of CBOW is to maximise the conditional probability of the output word given the input context, which is expressed as follows:

$$L(w_o) = -\log P(w_o \mid c(w_t)) = -\vec{w}_o{}^\top \cdot \vec{h} - \log\left(\sum_{v_i \in V} \exp\left(\vec{v}_i{}^\top \cdot \vec{h}\right)\right) \tag{2.8}$$

where $\vec{w}_o$ is the predicted output vector by the model. The model updates the weights of the hidden layer using the back-propagation algorithm and gradient descent.

2.3.3 The skip-gram model

The skip-gram model (also commonly referred to as skip-gram with negative sampling (SGNS)) is the second algorithm introduced by Mikolov *et al.* (2013a,b), as

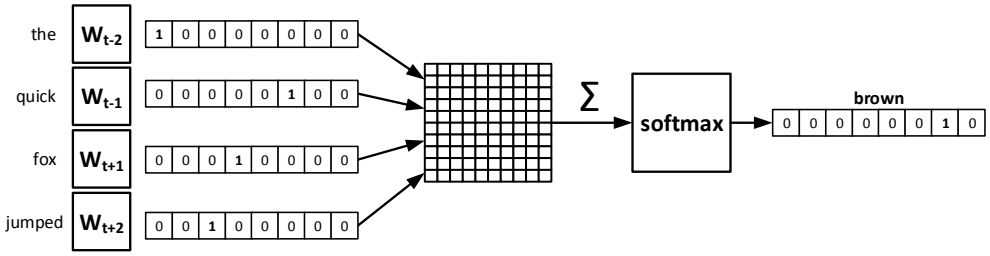


Figure 2.1: An illustration of the CBOW neural network model. The symbol 'Σ' denotes the averaging of the input context words ($c(w_t)$) vectors multiplied by the hidden layer weights. The SoftMax function estimates a probability distribution over all words in the vocabulary.

[15] Mikolov *et al.* (2013a,b) optimised this model by using hierarchical SoftMax and a binary Huffman tree to encode the vocabulary.

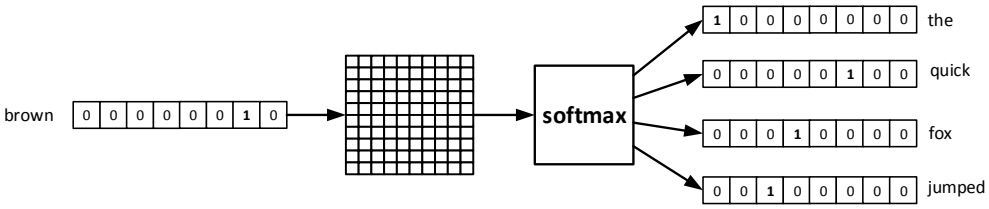


Figure 2.2: An illustration of the skip-gram model.

part of the `word2vec` tool. It is conceptually the reverse of the CBOW model, where now the target word is the input to the model, and the goal is to predict the context of the target word (as illustrated in Figure 2.2).

The skip-gram model shares much of the mechanics of CBOW, but with several key differences. While the hidden layer remains the same, its output $\vec{h}$ is no longer an average of context words—it is simply calculated as follows:

$$\vec{h} = \mathbf{H}_{(k,.)} = \vec{w}_t{}^{\top} \tag{2.9}$$

In other words, $\vec{h}$ is simply transposing a row of the hidden layer weight matrix that is associated with the model's only input one-hot vector. The SoftMax output layer is also modified such that it no longer calculates the probability distribution of all words in the vocabulary, but rather, only the words that appear in all seen contexts of the target word, *i.e.*, $c_i \in c(w_t)$, in order to predict the context words of the target word:

$$P(c_i \mid w_t) = \frac{\exp\left(\vec{c}_i{}^{\top} \cdot \vec{h}\right)}{\sum_{c_i \in c(w_t)} \exp\left(\vec{c_i}^{\top} \cdot \vec{h}\right)} \tag{2.10}$$

Likewise, the loss function is also modified to reflect the objective of the skip-gram model—to maximise the conditional probability of the context words given the target word:

$$\begin{aligned} L\left(c\left(w_t\right)\right) &= -\log P(c\left(w_t\right) \mid w_t) \\ &= -\sum_{c_i \in c(w_t)} \left(\vec{c_i}^{\top} \cdot \vec{h}\right) + \left|c\left(w_t\right)\right| \cdot \log\left(\sum_{v_i \in V} \exp\left(\vec{v_i}^{\top} . \vec{h}\right)\right) \end{aligned} \tag{2.11}$$

In order to compute the gradient for the SoftMax layer, the model needs to perform a summation across all classes, *i.e.*, all words in the vocabulary (the denominator of Equation 2.10). This is computationally expensive as there is typically a very large number of word vectors that need to be updated per iteration of training.

Mikolov *et al.* (2013a,b) use *negative sampling* as an optimisation that requires updating only a small sample of vectors. This is achieved by updating only the predicted context words (positive words) as well as a small random sample of negative words, *i.e.*, words that the model predicted not to appear in the context, rather than the entire vocabulary.

2.4 Standard evaluation metrics

In this section, we describe a set of standard evaluation measures used widely in NLP and the work presented herein. These measures are commonly used to quantify the performance of a supervised binary classification algorithm regardless of the task. The same measurements may still be used for multiclass classification by subdividing the evaluation into $|C|$ binary classifications, where C is the set of classes. The most basic performance metric is *accuracy* (A), which simply measures the proportion of instances that the classifier predicted correctly:

$$A = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} = \frac{\text{TP} + \text{TN}}{n} \tag{2.12}$$

where TP are the true positives, FP are the false positives, TN are the true negatives, FN are false negatives, and n is the number of example instances in the evaluation set.

The accuracy metric can be problematic as it is susceptible to unbalanced datasets (*i.e.* when either the positive or negative examples are greatly overrepresented in the data). For example, if a dataset has a very large number of negatives, a trained classifier may easily learn to classify negative examples but still poorly classify positive examples. Then, the number of TN will be high due to the nature of the data, and thus accuracy will also be high, even if TP is poorly predicted by the classier. In other words, the accuracy metric does not distinguish between type I[16] and type II[17] errors.

Precision (P) and *recall* (R) are used to help quantify type I and II errors, respectively. Precision measures the proportion of the positively classified instances that are correctly predicted by the classifier, while recall measures the proportion of positive instances in the data that the classifier is able to predict. They are defined as follows:

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2.13}$$

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2.14}$$

[16] A type I error is a false positive classification.
[17] A type II error is a false negative classification.

It is often convenient to combine both precision and recall using the harmonic mean; this measure is known as the F_1-*score*:

$$F_1 = \frac{2PR}{P+R} \tag{2.15}$$

The '1' in F_1 indicates that both recall and precision are weighted equally. It is possible to select different weight parametrisation for the F score—for example, $F_{0.5}$ which gives more weight to recall; conversely, F_2 favours precision. All mentions of F score in this work refers to the standard F_1 parametrisation.

In the case of multiclass[18] or multilabel[19] evaluation, it is often the case that an averaged metric is given across all of the classes using the mean. This is known as a *macro-average*: if C is the set of classes, then the macro-averaged metrics are as follows:

$$\bar{A} = \frac{\sum A_i}{|C|} \qquad \bar{P} = \frac{\sum P_i}{|C|} \qquad \bar{R} = \frac{\sum R_i}{|C|} \qquad \bar{F}_1 = \frac{\sum F_{1,i}}{|C|} \tag{2.16}$$

where the subscript i denotes the score for the ith class in C. Another type of averaging is a *micro-average*, where the individual components of these metrics[20] are summed up and then calculated as per the following formulae:

$$P_\mu = \frac{\sum \mathrm{TP}_i}{\sum \mathrm{TP}_i + \sum \mathrm{FP}_i} \qquad R_\mu = \frac{\sum \mathrm{TP}_i}{\sum \mathrm{TP}_i + \sum \mathrm{FN}_i} \qquad F_{1,\mu} = \frac{2P_\mu R_\mu}{P_\mu + R_\mu} \tag{2.17}$$

A binary classifier uses a decision threshold to decide whether to label an instance with a positive or negative label from a numeric score[21] it assigns each instance. A decision threshold of 0.5 may be used on a numeric output range of 0 to 1. If an instance is scored with less than 0.5, then it would be labelled a negative example, otherwise it would be labelled as a positive example. This threshold value is typically set equal to the proportion of positive to negative examples in the dataset (*i.e.* a threshold value of 0.5 for a balanced dataset), but sometimes it is fine-tuned on a development held-out data, or learned from data.

The metrics described thus far only assess the classifier's performance for exactly one decision threshold. In many situations, these scores do suffice, but it can sometimes be helpful to evaluate the classifier's performance with respect to all thresholds. This is typically achieved using the receiver operating characteristic

[18] *Multiclass* usually means that an instance is assigned exactly one class from a set that contains more than two mutually exclusive classes.

[19] *Multilabel* usually means that an instance can be assigned one or more labels (classes) from a set that contains mutually nonexclusive labels.

[20] The accuracy metric is identical using both macro- and micro-averaging.

[21] Typically a confidence probability value, but not necessarily, as in the case of SVMs.

(ROC) curve, which is a plot of the true positive rate (*i.e.* recall) against the false positive rate[22] (FPR = $\frac{\text{FP}}{\text{FP+TN}}$) at all decision thresholds of the classifier.

The area under receiver operating characteristic curve (AUC) provides a way to quantify the classifiers performance at every decision threshold, and is equal to the probability that the classifier will rank a randomly chosen positive example higher than a randomly chosen negative example. AUC is calculated by integrating the ROC curve with respect to the threshold as follows:

$$\text{AUC} = \iint_{-\infty}^{\infty} \mathbf{1}_{T'>T} \cdot f_0(T') \cdot f_1(T)\, \mathrm{d}T'\mathrm{d}T \tag{2.18}$$

where T is a variable that represents the threshold, f_0 is the probability density function that an instance is assigned a positive class (*i.e.* if the predicted numerical value is less than the threshold T); conversely, f_1 is the probability density function that an instance is negative.

There are several methods for numerically integrating for AUC. The following equation is one common way to estimate the area:

$$\widetilde{\text{AUC}} = \frac{1}{|C_0| \cdot |C_1|} \sum_{i=1}^{|C_1|} \sum_{j=1}^{|C_0|} \mathbf{1}_{p_1(x_i) > p_0(x_j)} \tag{2.19}$$

where C_1 and C_0 are the sets of instances assigned positive and negative labels respectively; $p_1(x_i)$ is the probability that the classifier assigns instance x_i a positive label, likewise $p_0(x_j)$ is the probability that the classifier assigns instance x_j a negative label.

2.5 Chapter summary

Scientific literature is one of the main resources used in BIONLP research. The nature of the language used in biomedical literature differs significantly from general literature. It is often dense with long sentences containing complex clausal structures, using very domain-specific terminology and acronyms. Tools intended for the biomedical domain require resources such as annotated corpora, lexicons, and terminologies that are often annotated by experts in their relevant domains.

Biomedical information extraction refers to the process of extracting information from free text. Typical IE tasks are biomedical named entities NER, relation extraction, and event extraction. Major applications of these tasks include pathway curation, ontology extraction, information retrieval, and text mining.

Text mining systems generally consist of four stages: information retrieval, concept recognition, relation (or association) extraction, and knowledge discovery.

[22] Also known as *fallout*.

SVMs are a supervised algorithm that learns a hyperplane (decision boundary) in a high dimensional feature space. The "kernel trick" maps low dimensional feature input space to a higher dimensional space which helps make classification problems more separable. CBOW is an unsupervised representation learning algorithm that has the objective of estimating the probability for words given the context; doing so, it learns a vector representation (embedding) for the target word. SGNS is conceptually the reverse of the CBOW model, where the target word is the input to the model and the objective is to predict its context.

Chapter 3

Semantic text classification for cancer research

Semantic text classification is a core task in NLP with widespread applications in the biomedical domain, such as TM (Hunter and Cohen, 2006). Semantic text classification is the task of assigning one or more predefined labels to a span of text, where the label captures some semantic or topical information of interest (Joachims, 2002). This differs from other forms of text classification such as such spam detection, language identification, and sentiment analysis where the labels assigned to the text do not represent semantic information (Zhang and Liu, 2007). If the scope of the text is large (such as a document), it is often referred to as *document classification*—for example, assigning Medical Subject Heading (MeSH) to biomedical publications (Lowe and Barnett, 1994). If the scope of task is narrow (perhaps a sentence or a phrase), it would be considered akin to *information extraction*—for example, classifying sentences into document structure (argumentative zones) (Agarwal and Yu, 2009).

This chapter introduces the two new text classification tasks for cancer that will form the basis of this work. First, semantic text classification is described in general, along with its application to the field of cancer by summarising related published work. Then, two new tasks are introduced: the Hallmarks of Cancer classification task and an exposure assessment (the Exposure taxonomy) classification tasks.

3.1 Related work

The two main kinds of data to which automatic text classification has been applied in the cancer domain are clinical reports and scientific publications. Some key publications relating to both of these types of data are briefly summarised here.

In the clinical domain, text classification systems have mostly focused on diagnosis. The input text is largely extracted from medical reports, electronic health records, or death certificates. Burnside *et al.* (2000) focus on classification of ra-

diology mammography reports into Breast Imaging Reporting and Data System (BI-RADS) assessment categories. They applied a linear least-squares fit mapping algorithm (Yang and Chute, 1992) to a very basic featureset based on word frequencies and terms from the BI-RADS lexicon. Later, they implemented a more advanced method (Nassif *et al.*, 2009) that is more accurate (97.7% precision, 95.5% recall, and 97% F_1-score).

Koopman *et al.* (2015) built a system that diagnoses several types of cancer. The system is trained on a corpus of death certificates using handcrafted features such as *n*-grams and the Systematized Nomenclature of Medical – Clinical Terms (SNOMED CT) (Stearns *et al.*, 2001). SVM classifiers were used in cascaded architecture: the first level identifies the presence of cancer and the second level identifies the type of cancer according to the International Statistical Classification of Diseases and Related Health Problems (10th revision) classification system (World Health Organization, 2009). The system can effectively diagnose common cancers (70% F_1-score) and rare cancers (12% F_1-score).

Within the cancer diagnosis domain, there are systems that focus on classifying text according to progression of the cancer it describes—*cancer staging*. This is also known as TNM classification—it is based on the extent of the primary **t**umour (T), spread to nearby lymph **n**odes (N) and distant **m**etastasis (M). For example, Nguyen *et al.* (2010) classify pathology reports according to the TNM framework by developing a symbolic rule-based classification system that exploits report substructure and the symbolic manipulation of SNOMED CT. On a corpus of pathology reports for 718 lung cancers, they achieve accuracies of 72%, 78%, and 94% respectively for T, N, and M staging. Another system is the Cancer Stage Interpretation System (CSIS) by McCowan *et al.* (2007), which classifies lung cancer stage based on T and N, but not M. CSIS uses SVMs with handcrafted featues. It achieves an accuracy of 74% for tumour (T) staging and 87% for node (N) staging.

Most other cancer-domain text classification systems use scientific literature, typically acquired through MEDLINE or `PubMed`. These systems tend to focus less on diagnoses, but rather carry out a wide variety of tasks.

Work by Lee *et al.* (2007) attempts to classify `PubMed` abstracts into six types of cancer: breast, cervical, gastric, lung, rectal, and œsophagal. Using SVMs and a dictionary of words as features, they achieve F_1-scores in the 80% to 90% range.

More recent work by Hughes *et al.* (2017) attempts to classify sentences from `PubMed` articles according to 26 medical categories from the Merck manual (Beers *et al.*, 1999) with some of the categories relating to the cancer domain (*e.g.* "brain cancer"). They use a CNN trained on 15,000 articles, and achieve an accuracy score around the 60% range.

CRA is another important area of work in semantic text classification for the cancer domain. TM was first introduced to CRA by Korhonen *et al.* (2009), who present a taxonomy to categorise the primary types of scientific evidence found in literature used by researchers attempting to determine carcinogenic properties of chemicals. Their taxonomy is divided into three components:

- **Carcinogenic activity**: various types of studies in found in literature. For example, "human study/epidemiology", "animal studies", "cell experiments", *etc.*.
- **Mode of Action**: groups key events (modes) leading to cancer. For example, "mutagenesis", "increased cell proliferation", and "receptor activation".
- **Toxicokinetics**: describes the process of uptake of chemicals by the body: the metabolism and biotransformation, and the distribution and excretion.

In addition to the taxonomy, Korhonen *et al.* (2009) also annotated a corpus based on 1297 PubMed abstracts classified according to the 48 classes of the taxonomy. Their work also applies supervised machine learning and achieves a good level of accuracy.

Korhonen *et al.* (2012) further develop their methodology by including additional features, and using a non-linear kernel in their SVM classifiers. They integrate their methodology with PubMed search as part the CRAB[23] TM tool. CRAB achieves a good level of performance: 78% micro F_1-score, an improvement over their previous methodology which achieved 76%. Guo *et al.* (2011, 2014) improve again the performance of CRAB by applying a combination of active learning and self-training. They improve the CRAB tool's macro-averaged F_1 score to 78%.

3.2 The Hallmarks of Cancer

This section introduces the task of semantic text classification according to the HoC.

The Hallmarks of Cancer (HoC) were first introduced in the seminal paper of Hanahan and Weinberg (2000), the most cited paper in the journal *Cell*. The paper introduced six hallmarks, which were then extended in a follow-up paper (Hanahan and Weinberg, 2011) to add a further four, thus forming the set of ten hallmarks that are known today.

Hanahan and Weinberg (2000) propose that all cancers share common traits—*hallmarks*—that govern the transformation of normal cells to malignant or tumourous cancer cells. By creating the set of hallmarks, they attempt to reduce the complexity of cancer to a small number of underlying principles. These Hallmarks of Cancer are used by cancer biologists, researchers, and oncologists as a way to easily describe and organise the properties of cancer (Lazebnik, 2010)

The Hallmarks of Cancer have not previously been proposed as an NLP task, and there are no corpora specifically for the hallmarks. However, they are still used in a number of key BIONLP works. The six original hallmarks were used as an organising principle in the BIONLP Shared Task 2013 Cancer Genetics task (Pyysalo *et al.*, 2013b). This task involved the extraction of events—biological processes—from cancer-domain texts. The hallmarks have also inspired other efforts

[23] http://omotesando-e.cl.cam.ac.uk/CRAB/request.html

in information extraction, and the development of tools such as *OncoSearch* (Lee, 2014) and *OncoCL* (Doland, 2014).

As the hallmarks are so widely used and recognised in cancer literature, we will largely focus on the classification of these hallmarks throughout this work.

The current set of hallmarks distils our knowledge of the disease into a fixed set of alterations in cell physiology. The ten hallmarks can be briefly described as follows:

- **Sustaining proliferative signaling**: Healthy cells require molecules that act as signals for them to grow and divide. Cancer cells, on the other hand, are able to grow without these external signals.
- **Evading growth suppressors**: Cells have processes that halt growth and division. In cancer cells, these processes are altered so that they don't effectively prevent cell division.
- **Resisting cell death**: Apoptosis is a mechanism by which cells are programmed to die in the event that they become damaged. Cancer cells are able to bypass these mechanisms.
- **Enabling replicative immortality**: Non-cancer cells die after a certain number of divisions. Cancer cells, however, are capable of indefinite growth and division (immortality).
- **Inducing angiogenesis**: Cancer cells are able to initiate angiogenesis, the process by which new blood vessels are formed, thus ensuring the supply of oxygen and other nutrients.
- **Activating invasion & metastasis**: Cancer cells can break away from their site of origin to invade surrounding tissue and spread to distant body parts.
- **Genome instability & mutation**: Cancer cells generally have severe chromosomal abnormalities, which worsen as the disease progresses.
- **Tumor-promoting inflammation**: Inflammation affects the microenvironment surrounding tumours, contributing to the proliferation, survival, and metastasis of cancer cells.
- **Deregulating cellular energetics**: Most cancer cells use abnormal metabolic pathways to generate energy. For example, they may generate energy *via* glucose fermentation even when enough oxygen is present to properly respire.
- **Avoiding immune destruction**: Cancer cells are invisible to the immune system.

In collaboration with cancer researchers at Karolinska Institutet, Sweden (widely regarded as one of the major research centres for medicine), we further expand the

original ten hallmarks to form a taxonomy of 37 classes that further specifies the biological processes underlying each hallmark. Figure 3.1 illustrates the extended taxonomy which consists of two levels: the first level contains ten primary classes representing the original hallmarks, and the second level consists of subclasses that represent more specific cellular or molecular processes. The subclasses are described in Appendix A.

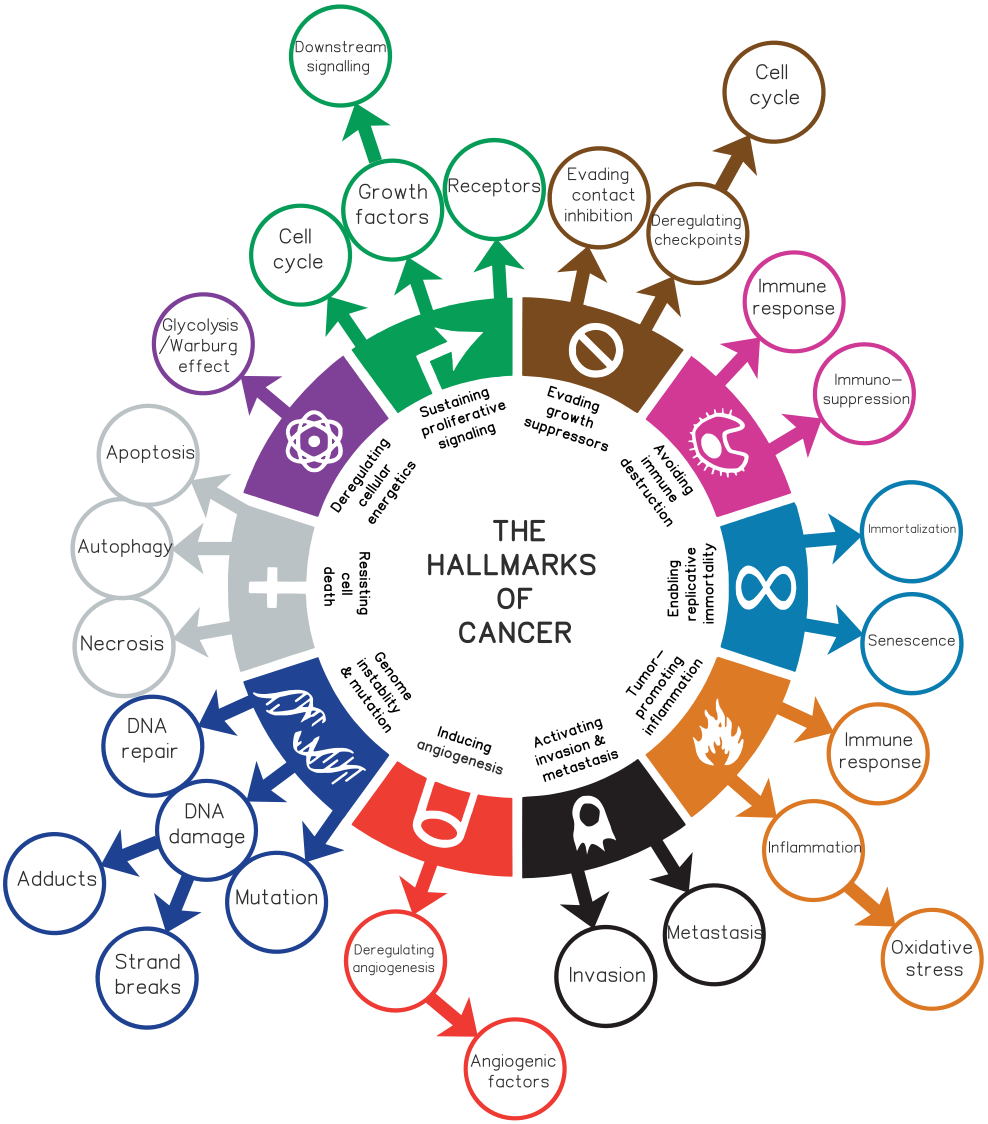


Figure 3.1: The Hallmarks of Cancer taxonomy. The inner circle represents the original ten cancer hallmarks described in Hanahan and Weinberg (2000, 2011), and the outer circles indicate the cellular processes associated with each cancer hallmark.

Our task is to classify text at the sentence and document levels according to one or more of the nodes in the proposed taxonomy of hallmarks. To facilitate this,

we annotated a corpus of PubMed abstracts according to the taxonomy—the HoC corpus. In the following sections, we describe this corpus, the NLP pipeline used to classify it at the PubMed abstract level, and the evaluation of our methodology.

3.2.1 Annotated corpus

In this section we describe the HoC corpus and annotation process. The corpus was annotated by cancer researchers at Karolinska Institutet.

The starting point was to define the scientific evidence for HoC. The primary resources were the two articles by Hanahan and Weinberg (2000, 2011) which describe examples of the cellular processes, proteins, and genes involved in individual hallmarks. For example, "apoptosis" can provide evidence for the "resisting cell death" hallmark, as does "caspase 3" because it is known to drive the apoptotic process. We also gathered additional evidence in literature during the annotation process. For example, articles studying specific cellular processes often also mention proteins or genes that can provide evidence for hallmarks. When needed, we used the *KEGG pathways in cancer* (Kanehisa *et al.*, 2008) to confirm a protein's function and its role in cell signalling.

Abstracts were retrieved from PubMed journals representing sub-areas of biomedicine relevant to cancer research (*e.g.* molecular biology, public health, and clinical medicine), using a set of search terms representative for each of the ten hallmarks. The terms and their synonyms appearing in Hanahan and Weinberg (2000, 2011) were employed, along with additional terms selected by our experts (see Table 3.1 for examples of the search terms). When needed, the term "cancer" was added to filter out irrelevant abstracts (*e.g.* those concerning the "immune response" without any obvious link to cancer). The PubMed searches were limited to the years 1992, 2002, and 2012 to ensure coverage of varied data over time. The total number of retrieved abstracts per hallmark ranged from less than a hundred to several thousand.

The annotation was conducted by an expert with over fifteen years of experience in cancer research. The annotation tool described by Guo *et al.* (2012) was used with its menu items customised to our hallmark task. The abstracts were chosen for annotation randomly, starting from the top of the list returned by PubMed search. Abstracts not containing information about hallmarks were left unannotated, as were those linked to review articles. Annotation was performed at sentence level so that only sentences describing findings or conclusions of the study in question were included. A sentence was annotated when it contained clear evidence for one or several hallmarks. In the latter case multiple labels were assigned to the sentence. Table 3.2 shows example labelled sentences, with evidence keywords highlighted.

The prior distribution of hallmarks is not uniform—some hallmarks occur more frequently in literature than others. This fact is reflected in the annotated corpus. Table 3.3 shows the number of annotated sentences and abstracts for each hallmark.

To investigate the accuracy of annotations, we performed inter-annotator agreement analysis where a second expert annotator was asked to annotate a subset of 4963 sentences which were then compared to those of the original annotator. We

Table 3.1: Hallmarks and their search terms.

Hallmark	Search term
1. Sustaining proliferative signaling (**PS**)	Proliferation Receptor Cancer "Growth factor" Cancer "Cell cycle" Cancer
2. Evading growth suppressors (**GS**)	"Cell cycle" Cancer "Contact inhibition"
3. Resisting cell death (**CD**)	Apoptosis Cancer Necrosis Cancer Autophagy Cancer
4. Enabling replicative immortality (**RI**)	Senescence Cancer Immortalization Cancer
5. Inducing angiogenesis (**A**)	Angiogenesis Cancer "Angiogenic factor"
6. Activating invasion & metastasis (**IM**)	Metastasis Invasion Cancer
7. Genome instability & mutation (**GI**)	Mutation Cancer "DNA repair" Cancer Adducts Cancer "Strand breaks" Cancer "DNA damage" Cancer
8. Tumor-promoting inflammation (**TPI**)	Inflammation Cancer "Oxidative stress" Cancer Inflammation "Immune response" Cancer
9. Deregulating cellular energetics (**CE**)	Glycolysis Cancer "Warburg effect" Cancer
10. Avoiding immune destruction (**ID**)	"Immune system" Cancer Immunosuppression Cancer

Table 3.2: Examples of sentences and keywords as evidence for annotated hallmarks.

Annotated hallmark	Examples of sentences with evidence (highlighted) for the annotated hallmarks
Sustaining proliferative signalling - cell cycle	Results indicate the PCNA labelling with PC10 is a simple method for assessing the proliferative activity in formalin-fixed, paraffin-embedded tissue of NSCLC and correlates well with Ki-67 labeling and S-phase fraction of the cell cycle.
Evading growth suppressors - cell cycle check points & contact inhibition	Subsequently, sod3-transduced MEF cells developed co-operative p21-p16 downregulation and acquired transformed cell characteristics such as increased telomerase activity, loss of contact inhibition, growth in low-nutrient conditions, and in vivo tumorigenesis.
Deregulating angiogenesis - angiogenic factors	Phosphorylated Akt and VEGF-A are involved in angiogenesis of gastric adenocarcinoma, and Akt activation may contribute to angiogenesis via VEGF-A upregulation.
Genomic instability and mutations - DNA repair	Incubation of BLM-treated cells dCF/dAdo resulted in significant inhibition of the repair of BLM-induced DNA SSB.
Activating invasion and metastasis - metastasis	Occurrences of metastases during γ-IR treatment accompanied induction of EMT markers, including increased MMP activity.

calculated the inter-annotator agreement using Cohen's Kappa κ. We found an agreement of $\kappa = 0.67$ for the ten hallmarks, and $\kappa = 0.61$ for the entire taxonomy, indicating a substantial level of agreement among our experts (Fleiss *et al.*, 2013; Landis and Koch, 1977).

Table 3.3: Number of annotated abstracts and sentences for each hallmark class.

Hallmark	# Abstracts	# Sentences
Activating invasion and metastasis	291	667
Invasion	168	282
Metastasis	168	317
Avoiding immune destruction	108	226
Immune response	77	152
Immunosuppression	47	70
Cellular energetics	105	213
Glycolysis / Warburg effect	98	195
Enabling replicative immortality	115	295
Immortalization	51	111
Senescence	76	185
Evading growth suppressors	242	368
Deregulating cell cycle checkpoints	154	254
Cell cycle	145	240
Evading contact inhibition	92	117
Genomic instability and mutation	333	771
DNA damage	182	372
Adducts	38	97
Strand breaks	71	123
DNA repair mechanisms	122	216
Mutation	102	215
Inducing angiogenesis	143	357
Deregulating angiogenesis	139	349
Angiogenic factors	77	170
Resisting cell death	430	833
Apoptosis	339	611
Autophagy	53	157
Necrosis	76	108
Sustaining proliferative signalling	462	993
Cell cycle	189	320
Growth factors growth promoting signals	177	323
Downstream signalling	92	138
Receptors	184	345
Tumor promoting inflammation	194	437
Immune response	45	78
Inflammation	178	371
Oxidative stress	80	158

Each sentence can be annotated with one or more labels. Therefore, it is important to see how many labels are assigned per instance in the database as a distribution. Figure 3.2 shows the distribution of abstracts and sentences according to the number of labels that they are assigned. We count each node in the taxonomy as a separate label, including any hypernymy relations (*i.e.* if an instance has some label x, then

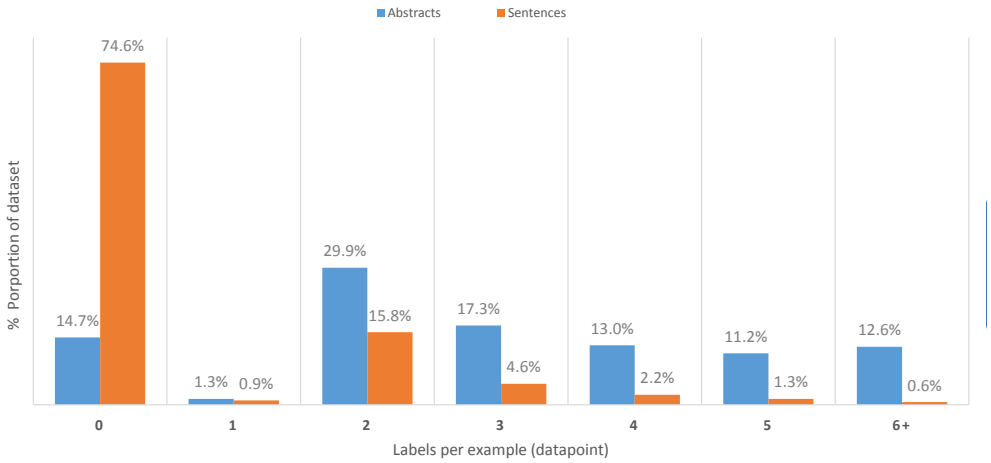


Figure 3.2: The proportion of examples (abstracts or sentences) in the HoC corpus according to the number of annotated labels per example.

it must also be annotated with all superclasses of label x).
The difference between the two distributions is evident in Figure 3.2—the vast majority (almost three quarters) of sentences are unlabelled, and on average abstracts will have more labels than sentences.

3.2.2 Methodology

This section details our initial methodology for automatically classifying text according to the HoC. The objective is to ascertain whether the annotated corpus is sufficient for standard supervised machine learning algorithms to reliably classify abstracts according to the hallmarks.

A conventional NLP pipeline (illustrated in Figure 3.3) that extracts a rich feature-set from text was used, in addition to using metadata retrieved from `PubMed` along with the corpus. The following describes each step of our pipeline.

The following describes each step of our pipeline:

1. **Data Cleaning**: We first pre-process data retrieved from `PubMed` by extracting text appearing in the abstract, automatically cleaning the text from possible encoding errors.

2. **Tokenisation:** We segment the text by words and then by sentences. We use BioTokenizer (Jiang and Zhai, 2007) to tokenise words and then Natural Language Tool Kit (Bird, 2006) for sentences. BioTokenizer achieves 96% Mean Average Precision, and is considered state of the art (Jiang and Zhai, 2007).

3. **Metadata Extraction:** Documents from `PubMed` are often accompanied by metadata capturing additional descriptive categorisation information about the document. We extract two types of information: the MeSH that are used

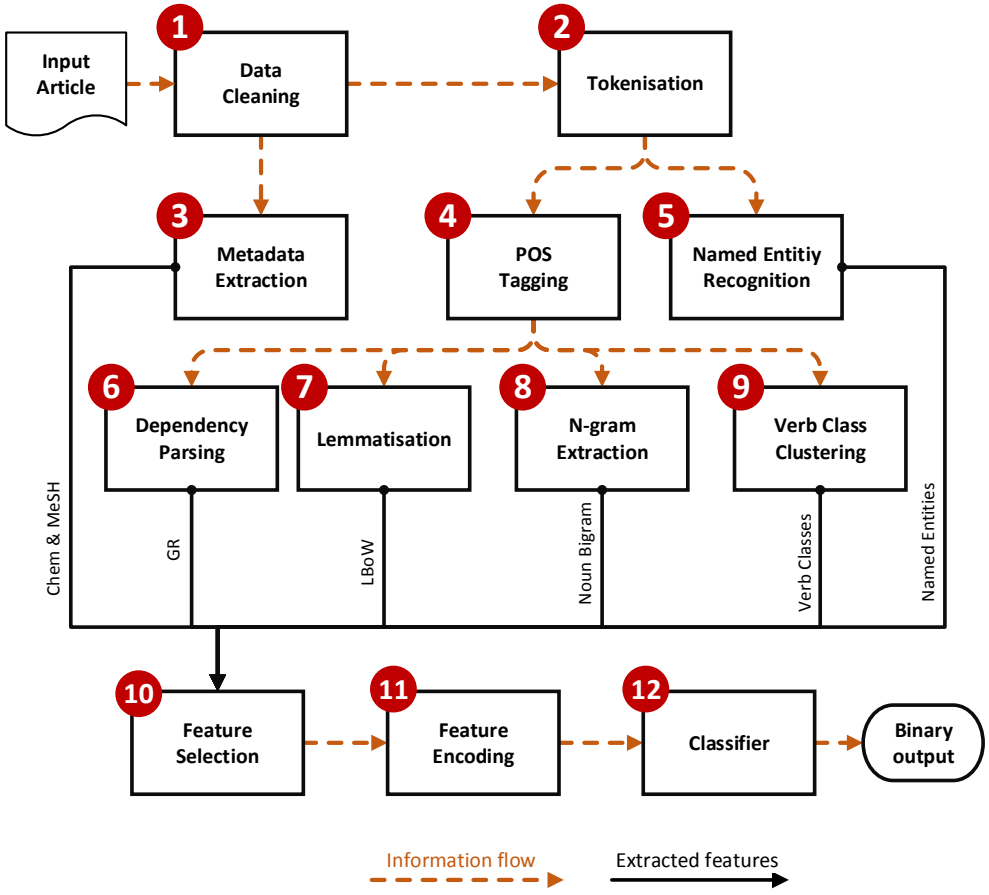


Figure 3.3: An illustration of our NLP pipeline.

to tag biomedical documents with subject categories assigned by professional annotators (Lowe and Barnett, 1994), and a list of chemicals (Chem) that are referred to in the article.

4. **POS Tagging:** POS tagging is the process of labelling each word in the text with a lexical category label. We use the *C&C tagger* (Clark, 2002; Clark *et al.*, 2002) which uses the Penn Treebank grammatical categories, and is trained on biomedical texts (Rimell and Clark, 2009).

5. **Named Entity Recognition:** We identify and extract named entities that are typically discussed in the biomedical literature. We extract five named entity types that are common in biomedical text: DNA, RNA, proteins, cell line and type. We store in the feature a pair of the entity type and the associated words or phrases. We use the NER tool `ABNER` (Settles, 2005), which is trained on the Natural Language Processing in Biomedical Applications and BioCreative corpora and achieves an F_1-score accuracy of 70.5% and 69.9% on these two

corpora respectively (Leitner *et al.*, 2010). By the end of this step, we extract the named entity (NE) feature type.

6. **Dependency Parsing:** We extract syntactic structures (trees) that encode grammatical dependency relations between words in the sentence, including direct object, non-clausal subject, and indirect object relations in parsed data taking into account their head and dependent words. We use the C&C Parser to extract these grammatical relations (GRs). We trained the C&C Parser using an available annotated biomedical corpus (Rimell and Clark, 2009). We stored each arc in the tree (for all sentences in the given abstract) in a list. This list was used to generate the lexicalised GR features, which are essentially template "patterns" that are matched during the feature encoding process. For example, a generated pattern could be ("subject <x>", "affect", "hemoglobin") for a simple *nsubj* relationship. If an abstract has this pattern, 1 is assigned for the corresponding index for this feature.

7. **Lemmatisation:** The goal of this stage of the pipeline is to produce Lemmatised Bag of Words (LBOW) features. A Bag of Words captures simply whether or not each of a set of words appearing in the corpus is found in a given abstract. We lemmatise (stem) the text in order to reduce sparsity of the words occurring. We use the *BioLemmatizer*, which is trained on biomedical texts (Liu *et al.*, 2012).

8. ***n*-gram Extraction:** We extract noun compound bigrams such as "breast cancer" or "gene silencing", as they can represent a concept in the text. We do not lemmatise the bigrams, as that could result in losing concept information (consider as an example "gene silencing").

9. **Verb class Clustering:** We group semantically similar verb predicates together. For example, verbs like "stimulate" and "activate" would be clustered together, whereas verbs like "halt" and "slow" would not. This allows us to generalise away from individual verbs and reduce data sparsity. We used the hierarchical classification of 399 verbs commonly used in biomedicine by Sun and Korhonen (2009). We use three levels of abstraction by allocating 3 bits in our feature representation for each concrete class (1 bit for each level of the abstraction hierarchy). By the end of this stage, we extract the verb class (VC) feature type.

10. **Feature Selection:** Features that are too rare or too common in the annotated corpus are removed, so that only the most discriminating features are used by the classifiers. The thresholds are set for each node by a process of trial and error, typically a minimum threshold value of 5 occurrences are selected, while the maximum threshold varies greatly depending on the feature type (usually larger than 100). This simultaneously improves accuracy and reduces training time. This procedure is applied separately for each node in the taxonomy, and

consequently each classifier has a unique set of selected features. Table A.2 details the number of features for each node after the feature selection step.

11. **Feature Encoding:** The features are represented in a sparse binary format, with a value of 1 indicating that the given abstract contains this feature. The feature itself can be as simple as a Bag of Words, or as complex as carefully engineered grammatical relations (if there is a certain dependency between specific word pairs), verb clusters (if there is a verb instance belonging to a given verb class), NER (if there is a specific word tagged with a specific named entity), *etc.*

12. **Classifier:** The binary features are then put into 37 binary classifiers (SVMs with radial basis function kernels) that label each abstract with a binary label indicating its relevance for one of the 37 nodes in the taxonomy. Each of the classifiers are trained and executed independently in order to have mutually non-exclusive multi-label classification.

    Each of the classifiers is trained on the entire corpus (unless we are evaluating the performance of the classifiers, where cross-validation is used). Abstracts that contain sentences annotated with its associated label are counted as a positive example for the training; otherwise, it is considered negative example. We use the hypernym–hyponymy relationships in our taxonomy to determine whether an example should be labelled positively or negatively for a given node (*i.e.* we count sub-node labels as positive examples when we are classifying abstracts under their parent node).

3.2.3 Intrinsic evaluation

We evaluate our classification methodology intrinsically by using standard measures of performance (precision, recall, accuracy, and F_1-score). We use nested cross-validation to avoid sampling bias, as recommended for small datasets (Statnikov *et al.*, 2008; Varma and Simon, 2006). The data is divided into four folds—the model is trained with 75% of the data and tested with the remaining 25%, and this split configuration is rotated four times for full coverage of the dataset. The size of folds was selected based on the sparsity of the test data. Within the 75% of the training data, we also perform another step of cross-validation for parameter tuning of the SVM kernels. Here we apply five-fold cross-validation, where we train with 80% of the data (for a given parameter configuration) and test on the remaining 20%.

Table 3.4 shows the results of the intrinsic evaluation. The performance of the classifiers varied from as low as 39% to as high as 94.1% F_1-score. On average, the performance was around 70% F_1-score (Macro: 68.4%, Micro 70%). The classifiers had substantially higher precision than recall for every node in the taxonomy (on average, precision is 14% higher than recall), the difference between precision and recall is more significant for nodes with lower F_1-score.

The reason for the variation in F_1-scores is that some categories are very sparse (as shown in Table 3.3), for example, the lowest scoring classifier (Immune response)

Table 3.4: The Hallmarks of Cancer taxonomy classification intrinsic evaluation results. All figures are expressed as percentages.

Hallmark	Precision	Recall	Accuracy	F_1-score
Activating invasion and metastasis	84.4	72.5	92.5	78.0
Invasion	70.7	51.8	92.6	59.8
Metastasis	69.1	56.0	92.7	61.8
Avoiding immune destruction	81.8	66.7	96.7	73.5
Immune response	64.3	46.8	96.1	54.1
Immunosuppression	73.3	46.8	97.9	57.1
Cellular energetics	96.6	81.9	98.6	88.7
Glycolysis / Warburg effect	91.6	77.6	98.2	84.0
Enabling replicative immortality	93.0	69.6	97.4	79.6
Immortalization	91.9	66.7	98.7	77.3
Senescence	91.5	71.1	98.3	80.0
Evading growth suppressors	65.1	58.7	88.9	61.7
Deregulating cell cycle checkpoints	53.2	43.5	90.8	47.9
Cell cycle	54.8	43.4	91.5	48.5
Evading contact inhibition	87.4	82.6	98.3	84.9
Genomic instability and mutation	87.8	77.5	93.0	82.3
DNA damage	80.6	68.7	94.5	74.2
Adducts	85.2	60.5	98.8	70.8
Strand breaks	74.1	56.3	97.2	64.0
DNA repair mechanisms	78.8	67.2	96.1	72.6
Mutation	74.3	53.9	95.8	62.5
Inducing angiogenesis	89.7	73.4	96.8	80.8
Deregulating angiogenesis	89.3	71.9	96.8	79.7
Angiogenic factors	77.2	57.1	97.1	65.7
Resisting cell death	84.9	76.0	89.8	80.2
Apoptosis	85.0	76.7	92.1	80.6
Autophagy	98.0	90.6	99.6	94.1
Necrosis	79.6	51.3	97.0	62.4
Sustaining proliferative signaling	72.2	61.7	81.8	66.5
Cell cycle	65.3	50.8	90.9	57.1
Growth factors growth promoting signals	61.0	48.6	90.8	54.1
Downstream signaling	54.1	35.9	94.5	43.1
Receptors	67.3	53.8	91.6	59.8
Tumor promoting inflammation	82.2	64.4	93.9	72.3
Immune response	53.8	31.1	97.3	39.4
Inflammation	77.8	59.0	93.5	67.1
Oxidative stress	79.6	53.8	97.0	64.2
Macro-average	77.5	61.5	94.7	68.4
Micro-average	77.7	63.7	94.7	70.0

Table 3.5: Leave-one-out feature analysis results (for classifying the Hallmarks of Cancer taxonomy). All figures are F_1-scores. Values in parentheses indicate that leaving out that feature improved the F_1-score.

Hallmark	All	LBOW	GR	NE	VC	NB	MeSH	Chem
Activating invasion and metastasis	78.0	68.7	(78.1)	77.6	(79.4)	76.2	76.9	(78.9)
Invasion	59.8	57.2	58.8	(60.3)	(60.1)	59.1	59.8	(60.1)
Metastasis	61.8	45.2	58.5	61.5	60.5	58.6	57.9	61.6
Avoiding immune destruction	73.5	54.0	70.4	72.4	72.4	73.0	69.0	72.7
Immune response	54.1	46.9	(55.2)	53.3	52.4	53.7	53.0	(54.5)
Immunosuppression	57.1	34.0	(61.5)	(60.0)	(62.3)	(68.3)	54.1	(57.9)
Cellular energetics	88.7	84.5	(88.8)	88.1	(89.8)	86.5	88.7	88.7
Glycolysis / Warburg effect	84.0	76.8	82.4	(84.2)	83.3	83.1	82.7	84.0
Enabling replicative immortality	79.6	69.9	(80.2)	79.0	(81.2)	(81.7)	78.4	79.0
Immortalization	77.3	61.4	(80.0)	77.3	(80.9)	(83.1)	73.6	(78.7)
Senescence	80.0	66.2	76.7	(81.8)	(84.9)	(81.9)	(81.8)	80.0
Evading growth suppressors	61.7	51.7	60.7	61.7	(63.0)	(63.8)	(61.9)	61.7
Deregulating cell cycle checkpoints	47.9	42.9	(48.8)	(48.9)	(50.7)	(52.1)	(50.7)	(48.4)
Cell cycle	48.5	44.7	48.3	(48.7)	(51.0)	(51.8)	(52.4)	(48.9)
Evading contact inhibition	84.9	77.3	82.7	84.9	84.4	84.9	83.8	84.3
Genomic instability and mutation	82.3	74.2	82.1	82.0	81.8	80.6	79.2	(83.0)
DNA damage	74.2	69.2	73.1	(74.6)	(74.9)	67.5	70.4	73.8
Adducts	70.8	50.0	67.7	70.8	67.7	72.7	74.6	70.8
Strand breaks	64.0	59.3	60.3	(65.1)	(64.1)	61.1	62.0	63.5
DNA repair mechanisms	72.6	71.4	(73.0)	72.6	74.2	71.0	69.9	71.4
Mutation	62.5	54.6	62.1	(62.9)	(65.2)	59.0	60.7	(64.0)
Inducing angiogenesis	80.8	70.7	(81.4)	(81.1)	79.1	77.6	79.4	(81.1)
Deregulating angiogenesis	79.7	71.9	(80.2)	(81.0)	77.6	79.1	78.1	(80.2)
Angiogenic factors	65.7	58.4	65.2	64.6	62.7	64.2	(68.1)	63.2
Resisting cell death	80.2	70.4	78.6	80.2	(80.5)	77.1	78.3	(80.4)
Apoptosis	80.6	70.0	(81.0)	80.5	80.4	78.8	79.7	(80.7)
Autophagy	94.1	82.5	91.8	93.1	93.1	89.6	86.0	94.1
Necrosis	62.4	28.4	(66.2)	(63.0)	(66.7)	(62.5)	57.4	61.3
Sustaining proliferative signaling	66.5	66.3	66.1	64.4	(67.1)	66.0	66.5	66.2
Cell cycle	57.1	56.8	55.7	56.7	56.8	53.7	(61.1)	(58.0)
Growth factors growth promoting signals	54.1	46.1	48.4	49.1	53.3	43.9	49.2	50.5
Downstream signaling	43.1	(45.2)	(44.7)	40.5	39.7	31.9	41.0	42.9
Receptors	59.8	57.3	58.1	59.5	(61.8)	55.0	(62.3)	(60.8)
Tumor promoting inflammation	72.3	60.5	69.7	71.1	(72.5)	68.7	68.0	71.6
Immune response	39.4	16.7	(40.0)	(42.9)	(50.7)	(40.0)	(40.0)	39.4
Inflammation	67.1	58.2	(67.5)	66.0	66.2	65.7	65.6	(68.4)
Oxidative stress	64.2	44.4	(65.2)	(65.7)	61.9	(65.3)	62.8	64.2
Macro-average	68.4	58.5	67.8	68.3	(69.0)	67.3	67.2	68.3

has only 45 positively labelled abstracts, a very small number for a classifier to reliably learn. It is important to note that the number of features for each classifier (Table A.2) is highly dependent on the number of positively labelled abstracts, for 'Immune response' node, there are only 375 features, the lowest of any node (The average number of features is 1114, as shown in Table A.2).

This also explains the difference between precision and recall, good level of recall is challenging to achieve in very sparse datasets (Davis and Goadrich, 2006), so for sparsely labelled nodes, recall is much lower than precision.

We also conducted leave-one-out analysis (feature ablation) in order to quantify the influence of each feature type on our classification performance for every node in the taxonomy. We achieve this be removing one of each of the feature types and train and cross-validate the classifiers on the remaining feature types.

Table 3.5 summarises the results of the leave-one-out feature analysis. We can see that on average that six of the seven feature types were produced lower F_1-score when removed, the exception being VC, where on average, the F_1-score improves very slightly by 0.6% when this feature is removed. The VC type benefits about 50% of the nodes, and is the weakest of all feature types. On the other hand, LBOW is the most influential feature type, and benefits all but one of the nodes in the taxonomy.

The leave-one-out feature analysis results show that on average the classifiers influenced more from lexical features (LBOW) than deeper linguistic features, such as grammatical relations and verb clustering. This perhaps is a reflection on the nature of the task, where word ordering and deeper syntactic structure is not as important as lexical surface features. This is more notable in some hallmarks, for example "Deregulating cell cycle checkpoints" which seems to benefit when in the analysis when each feature type is left out except for LBOW.

3.2.4 Case studies

We performed four case studies[24] to evaluate the usefulness of the hallmark classification on unseen data. In the first two, we apply our approach to literature on selected tumour types and anticancer drugs. The idea is to examine the ability of our method to "profile" a tumour type or a drug based on hallmark-classified literature and to compare the two profiles to see whether differences and similarities confirm the existing scientific knowledge.

For well-studied tumour types and drugs, the most relevant and frequent hallmarks are known by experts. Whether the automatically generated literature distribution profiles confirm this existing knowledge can be a good indicator of the reliability of the classification and can complement intrinsic system evaluation. The results of these case studies were tested for statistical significance using χ^2 homogeneity test for each hallmark (using a 2×2 contingency table) followed by a Bonferroni correction for the entire profile's p-values.

[24] Case studies 1 and 2 were designed by two experts in Karolinska Institutet and implemented by the author. Case studies 3 and 4 were designed and implemented by the author.

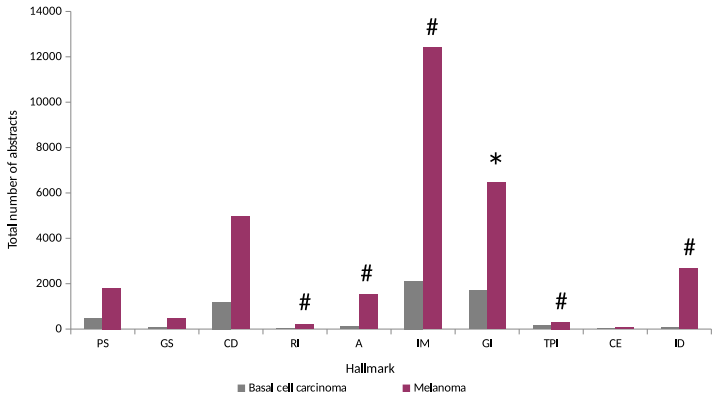


Figure 3.4: Case study 1: the distribution of Basal cell carcinoma and Melanoma literature over the relevant hallmarks. (*) denotes statistical significance level ($p < 0.05$), (#) denotes ($p < 0.001$).

In the last two case studies, we evaluate our approach in the context of information retrieval. As described earlier, the hallmarks do not normally appear explicitly as literal strings in text. Rather, they are latent in nature and retrieval of a comprehensive set of articles relating to these hallmarks requires using a large number of keywords like those presented in Table 3.1, and can result in a large number of false positives. Therefore, our goal is to show that we can identify a higher number of true instances than is realistic by simply using a standard keyword search, whilst keeping the number of false positives lower.

Case study 1: Basal cell Carcinoma *vs.* Melanoma

Basal cell carcinoma and melanoma are two types of human skin cancers with different biology, and consequently differing degrees of malignancy. Melanoma is highly metastatic with high mortality, while the more common basal cell carcinoma rarely or never metastasises and has lower mortality (Tomasetti and Vogelstein, 2015).

All `PubMed` abstracts available in December 2014, including 22,564 abstracts for basal cell carcinoma and 98,924 for melanoma, were used. The results of classification are shown in Figure 3.4. Out of a total of 121,488 abstracts from the original literature search only 46,727 abstracts (38%) were classified as relevant, highlighting the time-saving function of automatic classification.

Comparing the literature distribution for the hallmarks "activating invasion and metastasis", a significant difference can be seen with higher numbers of abstracts for melanoma. This reflects the existing knowledge about the metastatic potential of melanoma (Akinci *et al.*, 2008; Fidler, 1995; Young *et al.*, 2008). A significantly higher number of abstracts for melanoma were also found for "angiogenesis" and "avoiding immune destruction"—two parameters that further reflect malignancy.

This classification pattern is in line with existing scientific knowledge, demonstrating the reliable performance of our approach. Our methodology structures a large amount of textual information (more than 121,000 abstracts for the two tumour types in the original PubMed search) according to hallmarks—a task that would be almost impossible to conduct manually.

Case study 2: Sorafenib *vs.* Taxol

Sorafenib and Taxol are two drugs that have been developed to treat cancer *via* different mechanisms. Sorafenib acts by inhibiting development of new blood vessels (anti-angiogenic) (Wilhelm *et al.*, 2006), while Taxol inhibits cancer cell growth by inducing genomic instability (Schiff and Horwitz, 1980) .

We used for investigation all the PubMed abstracts available in December 2014, including 3,846 abstracts for Sorafenib and 24,827 for Taxol. The results of classification are illustrated in Figure 3.5. Out of a total of 28,673 abstracts in the original literature retrievals, only 8,993 abstracts were classified as relevant for cancer hallmarks (31%)—again highlighting the time-saving aspect of automatic classification.

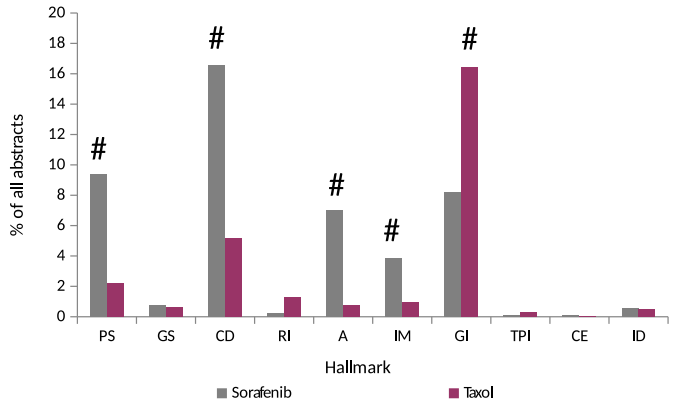


Figure 3.5: Case study 2: the distribution of Sorafenib and Taxol literature over the relevant hallmarks. (*) denotes statistical significance level ($p < 0.05$), (#) denotes ($p < 0.001$).

Comparing the literature distribution for Sorafenib and Taxol, there is a significant difference in the percentage of abstracts relevant to the aforementioned hallmarks: a significantly higher number of abstracts for Sorafenib were classified for the hallmarks "Inducing angiogenesis", "Resisting cell death" and "Sustaining proliferative signaling". This is in line with the anti-angiogenic effect of Sorafenib and its effect in causing cell death (Wilhelm *et al.*, 2006). The most frequent hallmark in Taxol literature is "genomic instability and mutation". This corresponds to existing knowledge about Taxol as a drug that interferes with microtubules and chromosomal segregation in a dividing cell, and may lead to genetic instability (Abal

Table 3.6: The search queries used to describe the ten hallmarks while searching for the topic: Melanoma.

Hallmark		Search query
5.	A	melanoma AND angiogenesis
3.	CD	melanoma AND "cell death"
9.	CE	melanoma AND "warburg effect"
7.	GI	melanoma AND "genomic instability mutation"
2.	GS	melanoma AND "growth suppression"
10.	ID	melanoma AND "immune destruction"
6.	IM	melanoma AND "invasion metastasis"
1.	PS	melanoma AND proliferation
4.	RI	melanoma AND immortalization
8.	TPI	melanoma AND inflammation

et al., 2003; Pihan and Doxsey, 1999). This study, again, reflects the accuracy of the literature distribution profiles and the reliability of the classification results.

Case study 3: Improving information retrieval

This case study investigates whether our classification approach can identify a higher number of relevant abstracts, with fewer false positives than a standard keyword search approach.

We compare the number of articles retrieved by `PubMed` keyword search to what our classifiers can identify for a given search topic. For our test topic, we use "Melanoma". We combine the search query "Melanoma" with a single search string according to expert's best description of each hallmark name (see Table 3.6 for the search queries used).

We can estimate the percentage of its false positives by ascertaining the relevance of the top 20 `PubMed`-retrieved search results for each hallmark using expert evaluation: an expert in cancer research is asked whether each of the top 20 retrieved abstract is really related to the given hallmark based on the abstract text. We can compare our classifiers' performance by retrieving articles from `PubMed` for "Melanoma" (98,924 abstracts in total), and then run our classifiers over these articles to find which of the "Melanoma" abstracts that are also associated with each of the hallmarks; therefore, we are identifying abstracts that are both relevant to the search topic "Melanoma" and each of the hallmarks, thereby directly comparing retrieval performance with that of the `PubMed` keyword search queries (Table 3.1).

We then evaluate the output of our classifiers for the same list of 20 abstracts per hallmark against our expert's judgement, and compare the percentage of false positives from our classifiers to the `PubMed` keyword search.

Table 3.7: Case study 3 results, comparing the number of abstracts retrieved from `PubMed` using the search queries (Table 3.6), and the number of classified abstracts out of a total of 98,924 abstracts using our approach. The % false-positive numbers are only of the top 20 retrieved abstracts using each of the search strings in Table 3.1, and not of the entire result set.

Hallmark		Keyword search		Our approach	
		# Retrieved	False-positives (%)	# Classified	False-positives (%)
5.	A	2155	0	1514	0
3.	CD	1813	0	4972	0
9.	CE	20	15	80	5
7.	GI	101	10	6478	0
2.	GS	105	35	472	0
10.	ID	35	0	2674	0
6.	IM	1954	0	12424	0
1.	PS	6958	0	1808	0
4.	RI	23	25	198	5
8.	TPI	1395	15	313	0
Average		1456	10	3093	1

Table 3.8: Search queries used by an independent user to retrieve documents about Melanoma, relating to each of the ten hallmarks.

Hallmark		Search query
5.	A	melanoma and "angiogenic factor"
3.	CD	melanoma AND apoptosis
9.	CE	melanoma AND glycolysis
7.	GI	melanoma AND "DNA damage"
2.	GS	melanoma AND "cell cycle"
10.	ID	melanoma AND immunosuppression
6.	IM	melanoma AND EMT
1.	PS	melanoma AND "growth factor"
4.	RI	melanoma AND telomerase
8.	TPI	melanoma AND "oxidative stress"

Case study 4: Information retrieval with an independent expert

In the previous case study, we constrained the keyword search string to terms that best describe the ten hallmarks. In this case study, we replicate the experimental setting from the previous case study, except that we do an unconstrained direct experiment where an independent expert in cancer research is asked to search for papers about the topic (melanoma) with a set of terms that they believe are associated with each of the hallmarks. The search terms do not need to be a description of the names of the hallmarks as in the previous case study, instead it is left to the expert to openly decide on any associated terms to search. Table 3.8 lists the resulting search queries selected by our expert volunteer.

The results show that overall, our classifiers identify substantially more abstracts

Table 3.9: Case study 4 results, comparing the number of abstracts retrieved from PubMed using the search queries in Table 3.8, and the number of classified abstracts out of a total of 98,924 abstracts using our approach. The % false-positive numbers are only of the top 20 retrieved abstracts using each of the search strings in Table 3.8, and not of the entire result set.

Hallmark		Keyword search		Our approach	
		# Retrieved	False-positives (%)	# Classified	False-positives (%)
5.	A	89	0	1514	0
3.	CD	4834	0	4972	0
9.	CE	142	35	80	5
7.	GI	868	0	6478	0
2.	GS	2512	0	472	0
10.	ID	1399	0	2674	0
6.	IM	1812	0	12424	0
1.	PS	3079	30	1808	0
4.	RI	191	85	198	5
8.	TPI	395	70	313	0
Average		1532	20	3093	1

than the keyword based approach for the majority of the hallmarks, while having a much lower percentage of false positives for the sample of top 20 retrieved results as compared to the keyword-based search.

One should note the very high false-positives for the hallmarks RI and TPI for the top 20 retrieved PubMed keyword search. This highlights some of the weaknesses of a keyword-only search approach, where even expert users may not find the optimal search terms for each hallmark. It shows the need to carefully select the correct hallmark-related terms in order to avoid false positives in a standard keyword search. In contrast, our classifiers can mitigate this risk since they take into account thousands of linguistic features instead of a small set of search terms.

3.2.5 Discussion

The intrinsic evaluation (Section 3.2.3) demonstrates the accuracy and the practical potential of hallmark-based text classification. We report here further analysis to gain insight into the errors made by the classifier and how to improve our approach in the future. We first looked into multi-labelling. As Table 3.10 shows, 40% of the abstracts in our annotated corpus are multi-labelled. Figure 3.6 displays the proportion of abstracts in the corpus according to the percentage of matched hallmark labels for a given abstract in the classifier output. 63.3% of abstracts have 100% of their labels correctly predicted by the classifier, while 14.8% have no matches (0% of their labels). The high percentage of 100% matched and 0% matched can be attributed to 60% of the abstracts having a single label.

We next examined the actual hallmark pair co-occurrences (i) in the annotated corpus (Table 3.10 where the diagonal line shows the number of occurrences for a given hallmark in the corpus independent of other co-occurring hallmarks) and

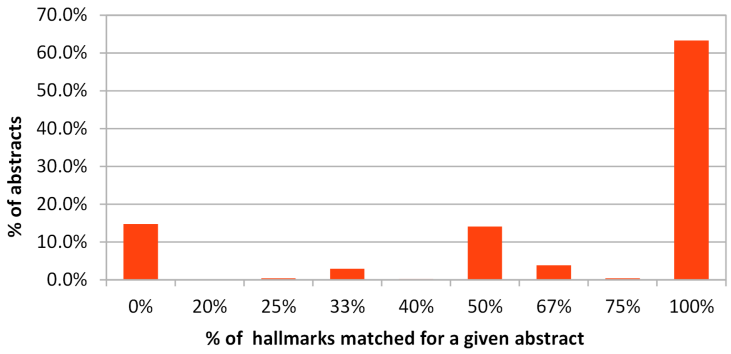


Figure 3.6: The distribution of all abstracts according to the percentage of their correctly predicted hallmarks

(ii) as predicted by the classifier (Table 3.11). Looking at Table 3.10, hallmarks that most often co-occur with each other include PS, CD, and GS; 165 abstracts are labelled as PS and CD, and 120 abstracts labelled as PS and GS. These co-occurrences could be explained by the fact that they are all related to cell cycle regulation. For example, in the sentence: "*Moreover, harmine not only induced endothelial cell cycle arrest and apoptosis, but also suppressed endothelial cell migration and tube formation as well as induction of neovascularity in a mouse corneal micropocket assay*", the phrase "cell cycle arrest" is a good indicator of PS and GS, and the word "apoptosis" is a good indicator of CD. This might be explained by overlapping capabilities (*e.g.* cell growth) and is likely to be the main reason for the lower classifier performance figures for these three hallmarks in Table 3.4. Looking at Table 3.11, for many hallmarks the predicted co-occurrences are well-correlated with those in the annotated corpus. For example PS and CD are co-classified 154 times which is comparable to their 165 co-occurrences in the corpus. Similar observations can be made with regard to PS and GS. Our current approach is based on training ten independent, binary classifiers to predict whether an abstract belongs to a given hallmark category.

The first two case studies provide additional evidence that our system correctly classifies literature over the HoC. The automatic system rapidly generated profiles that would have been difficult and very time-consuming to produce manually, which would facilitate overviews of scientific literature. In the future, the approach may be further developed to support the detection of novel patterns and research hypotheses in literature.

The last two case studies show that our approach can support information retrieval in comparison with a search string intersection query where the goal is to identify documents for a given topic, as well as articles that relate specifically to certain hallmarks. Our approach generally identifies more documents, and has a smaller percentage of false positives than standard keyword based search. This can perhaps be explained by the latent nature of the hallmarks in texts—the fact that they are rarely stated explicitly but rather *via* indirect correlation of terms

that describe relevant biological processes, and therefore are not easily found by basic keyword search. Our case studies also demonstrate that experts selecting the wrong set of search terms may result in a high number of false positives, and that our classifiers are not susceptible to this problem since they are trained on a large number of features, and not on the occurrence of a single search term.

Our analysis also suggests that many hallmarks could be sub-divided, for example, according to the established pathways involved in tumor development. During cancer development aberrantly regulated intracellular signaling pathways tend to rearrange networks regulating cancer cells and the networks themselves can be divided into sub-circuits which regulate certain capabilities of cancer cells, *e.g.* viability circuit.

Table 3.10: Hallmark co-occurrence distribution in the annotated corpus.

	PS	GS	CD	RI	A	IM	GI	PI	CE	ID
PS	**462**	120	165	19	38	79	31	25	9	5
GS	120	**242**	86	15	10	28	31	9	2	0
CD	165	86	**430**	23	28	48	44	37	16	14
RI	19	15	23	**115**	2	6	28	4	3	2
A	38	10	28	2	**143**	42	0	14	2	3
IM	79	28	48	6	42	**291**	14	24	9	13
GI	31	31	44	28	0	14	**333**	27	7	6
PI	25	9	37	4	14	24	27	**194**	7	14
CE	9	2	16	3	2	9	7	7	**105**	0
ID	5	0	14	2	3	13	6	14	0	**108**

Table 3.11: Hallmark co-occurrence distribution as predicted by the classifier

	PS	GS	CD	RI	A	IM	GI	PI	CE	ID
PS	**285**	110	154	14	32	64	23	14	9	8
GS	110	**142**	82	6	8	24	27	3	2	0
CD	154	82	**327**	14	21	37	38	29	13	12
RI	14	6	14	**80**	1	7	26	4	3	1
A	32	8	21	1	**105**	32	1	10	2	2
IM	64	24	37	7	32	**211**	12	15	7	11
GI	23	27	38	26	1	12	**258**	19	6	5
PI	14	3	29	4	10	15	19	**125**	6	16
CE	9	2	13	3	2	7	6	6	**86**	0
ID	8	0	12	1	2	11	5	16	0	**72**

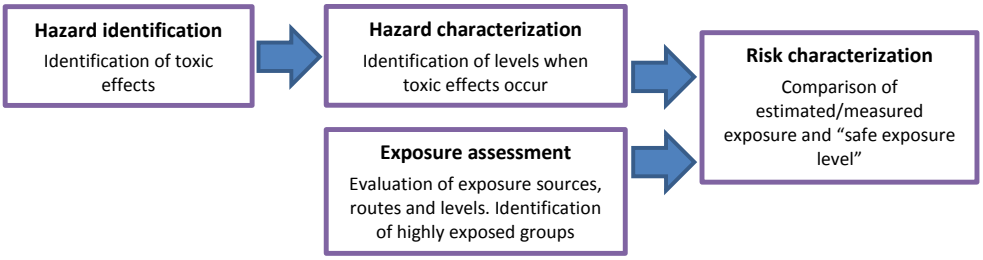


Figure 3.7: The process of chemical risk assessment includes exposure assessment, hazard identification, hazard characterisation and risk characterisation (FAO/WHO, 1995; NRC, 1983).

3.3 The Exposure Taxonomy

In this section we introduce a new text classification task with important relevance to cancer. Unlike the HoC taxonomy, which classifies text according to well known traits and biological processes that describe the inception and proliferation of cancer, we now focus on risk assessment. More specifically, this task is relevant on the previous work in cancer chemical risk assessment by Guo *et al.* (2014).

Korhonen *et al.* (2012, 2009) introduced the task of semantic text classification for chemical risk assessment, which uses a taxonomy that classifies evidence found in literature according to modes of action, types of study, and toxicokinetics. These are important factors for the process of risk assessment, but they fail to address another important aspect: exposure assessment, which is the focus of a new task that we introduce in this section.

Humans are constantly exposed to a large number of chemicals present in food, water, air, dust, soil, and consumer products *via* ingestion, inhalation, and dermal absorption. Some of these chemicals have known or suspected toxic effects that can cause disorders and diseases such as cancer at certain exposure levels. To estimate whether a population may be at risk at the current levels of exposure, risk assessments of chemicals are performed.

One cornerstone of the risk assessment process is the exposure assessment in which the magnitude, frequency, and duration of exposure are estimated or measured.

Exposure assessment methods include both indirect methods, such as exposure modelling and exposure calculations based on environmental measurements and questionnaire data, and direct measurements, such as human biomonitoring (HBM) and personal monitoring. HBM is the measurement of exposure biomarkers (chemicals or chemical metabolites) and effect biomarkers (indicators of effects caused by chemical exposure) in human body tissues or fluids (such as blood, hair, and urine).

An exposure taxonomy was developed by experts in biomonitoring and exposure sciences at Karolinska Institutet to categorise data into relevant classes. The taxonomy only concerns human exposure data, so animal and *in vitro* data are not

included. The taxonomy includes a total of 32 nodes divided into two main branches of exposure information: *biomonitoring* and *exposure routes*.

The first main branch categorises biomonitoring data and is further divided into exposure biomarkers (*i.e.* chemicals or chemical metabolites measured in human tissues to assess exposure) and effect biomarkers (*i.e.* markers of effects caused by chemical exposures). The exposure biomarker branch is further structured based on the biological matrix (*e.g.* blood, urine, hair/nail) in which the biomarker has been measured. The effect biomarker branch is further structured based on the character of the marker (*i.e.* molecule, gene, oxidative stress marker, other effect biomarker, or physiological parameter).

The second main branch contains data about the studied exposure routes—oral intake, inhalation, dermal exposure, and combined exposure routes. Table 3.12 provides a brief description of main classes in the Exposure taxonomy.

3.3.1 Annotated corpus

Two expert researchers from Karolinska Institutet annotated a corpus of abstracts in scientific journals representing a variety of relevant research areas. Seventeen well-studied compounds or groups of compounds representing different exposure routes and sources were selected for the first annotation round. These compounds were bisphenol A, phthalates, polychlorinated biphenyls, polybrominated diphenyl ethers, perfluorooctanesulfonic acid, chlordane, cadmium, arsenic, mercury, acrylamide, triclosan, naphthalene, benzene, styrene, toluene, particulate matter 2.5 and chlorpyrifos.

First, abstracts published between 1993 and 2013 were selected using a search term including the 17 chemicals mentioned above and 24 journals within in the fields of epidemiology, toxicology, exposure and environmental sciences. Because the exposure taxonomy only concerns human exposure data and not *in vitro* and animal studies, only abstracts indexed with the MeSH "humans" in `PubMed` were included in the annotated corpus. The search generated 5014 abstracts that were reviewed by an expert in exposure sciences and classified according to the taxonomy. The abstracts were only annotated if considered relevant for the taxonomy and consequently 2098 abstracts were annotated. As with the HoC corpus, annotation was performed at sentence level using the annotation tool by Guo *et al.* (2012).

After the initial broad search and annotations, several sub-nodes contained an insufficient number of annotated abstracts. Thus, additional searches using targeted keywords were performed. Some of these additional search strings included other journals and chemicals than the ones used the initial searches. The targeted searches generated 2748 abstracts, out of which 1588 abstracts were relevant and annotated.

In total, the annotated text corpus included 3686 annotated abstracts. In addition, 4076 abstracts were reviewed but considered irrelevant and were not annotated. Table 3.13 shows the number of annotated abstracts for each node in the taxonomy.

The distribution of annotation per example instance (abstract or sentence) is shown in Figure 3.8. Similar to the HoC corpus, there is large variation between the

Table 3.12: Description of the main classes (nodes) in the Exposure taxonomy.

Class label	Description
Biomonitoring	Biomonitoring data in humans.
Exposure Biomarker	
Blood, urine, hair/nail, adipose tissue, breast milk, placenta, other tissue.	Measurements of exposure biomarkers in human matrices.
Effect Marker	
Gene, molecule, other oxidative stress marker.	Measurements of effect biomarkers in human matrices.
Physiological	Measurements of physiological markers of effect, such as blood pressure, lung function, birth weight, *etc.*
Exposure Routes	Individual or population based exposure assessments.
Dermal uptake	Tape strip samples, hand wipes, hand washing samples, dermal wipes. Dermal exposure modelling.
Combined	Exposure modelling (*e.g.* PBPK) of several exposure routes simultaneously. Job exposure matrix. Intake calculations based on biomonitoring data.
Oral intake	
Food	Exposure from food (*e.g.* intake assessments, total diet studies, double portions, questionnaire data, *etc.*). Children exposed via breast milk.
Drinking water	Exposure from drinking water, bottled water, well water, *etc.*
Products	Exposure from toys, cosmetics, personal care products, dental amalgam, drugs and vaccines, household pesticides, *etc.*
Soil	Exposure from playground soil, residential garden soil, *etc.* Consumption of clay.
Dust	Exposure from dust in indoor microenvironments (homes, workplaces, schools, cars, *etc.*).
Inhalation	
Modelled exposure	Studies linking ambient air monitoring data with residential addresses (*e.g.* land regression models).
Other	Indoor air in indoor microenvironments (homes, workplaces, schools, cars *etc.*). Second hand tobacco smoke. Inhalation from showers, incense, cooking fuel, *etc.*
Personal measurement	Personal air monitoring. Breathing zone measurements.

Table 3.13: Number of annotated abstracts and sentences in the Exposure taxonomy corpus.

Class	# Abstracts	# Sentences
Biomonitoring	2621	7213
Effect marker	1201	2302
Biomarker	549	996
Gene	141	244
Molecule	426	695
- Lipid	94	139
- Other molecule	168	199
- Protein	300	434
Other effect biomarker	65	91
Oxidative stress	62	99
Physiological parameter	777	1342
Exposure biomarker	2206	5335
Adipose tissue	88	175
Blood	973	2077
Hair/nail	473	1134
Breast milk	177	377
Other tissue	143	299
Placenta	60	136
Urine	867	1985
Exposure Routes	2401	6199
Combined	165	218
Dermal Exposure	179	378
Inhalation	966	2221
Modeled exposure	296	612
Other inhalation	362	720
Personal air	230	462
Oral intake	1482	3673
Drinking water	450	958
Dust	289	753
Food	732	1625
Products	194	400
Soil	152	304

abstract and sentence distributions. The vast majority of sentences (65.9%) are not annotated with any labels, and abstracts tend to have more labels, with the majority are annotated with over five or more labels.

We performed inter-annotator agreement analysis where a second expert annotator was asked to annotate a subset of 200 abstracts. The annotation was compared to that of the annotator of the entire corpus. Near excellent agreement was found between the two annotators with the average Cohen's Kappa $\kappa = 0.79$ across the full taxonomy.

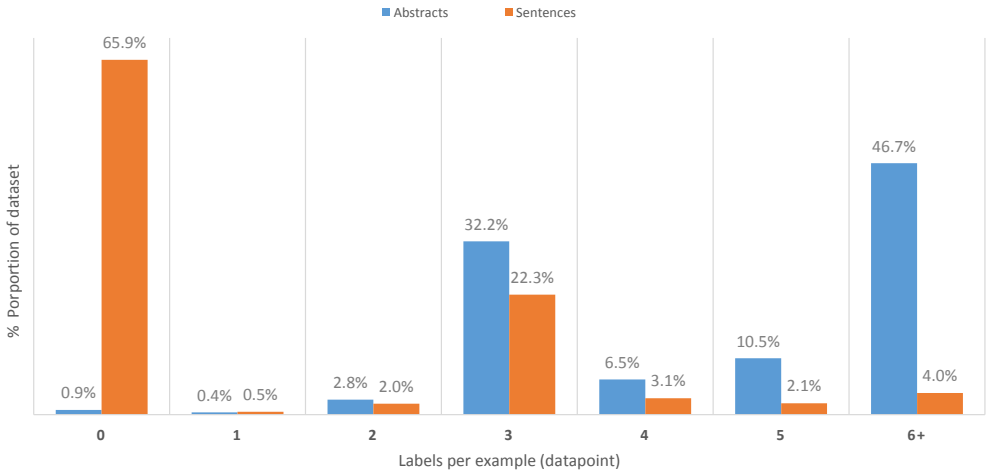


Figure 3.8: The proportion of examples (abstracts or sentences) in the exposure taxonomy corpus according to the number of annotated labels per example.

3.3.2 Methodology

We use exactly the same methodology for the Exposure taxonomy classification task as the HoC (Section 3.2.2) in terms of NLP pipeline and classification features and training setup. In Appendix A, Table A.3 shows the number of features for each label and feature type after the feature selection stage of the pipeline.

3.3.3 Intrinsic evaluation

We intrinsically evaluate our classifiers using the same protocol described in Section 3.2.3. This intrinsic evaluation is summarised graphically in Figure 3.9 with colour coding based on F_1-scores. The results are shown in Table 3.14.

The F_1-scores were generally high for the three top levels in the taxonomy. However, subdivision of effect biomarkers into distinct markers (*e.g.* gene, molecule, or oxidative stress marker) generally resulted in lower F_1-scores. Additionally, the nodes concerning indoor air, other effect biomarker, and other tissue for exposure biomarker measurements performed poorly.

We investigate the influence of each feature type on the classification accuracy. We conduct a leave-one-out feature analysis, where we repeat the experimental setup (described in the previous section), but with the removal of one of the seven feature types in order to ascertain its influence on the classification decision boundary for each node in the taxonomy. Table 3.15 details the results of this analysis.

On average, LBOW features have the most significant influence on the classification accuracy as it results in the largest drop (6.9%) on the averaged F_1-score. With the exception of only one category (other effect biomarker), all nodes benefit from the inclusion of LBOW features. On the other hand, the analysis shows that the removal of NE features marks the lowest drop (1.8%) in accuracy, with 6 out

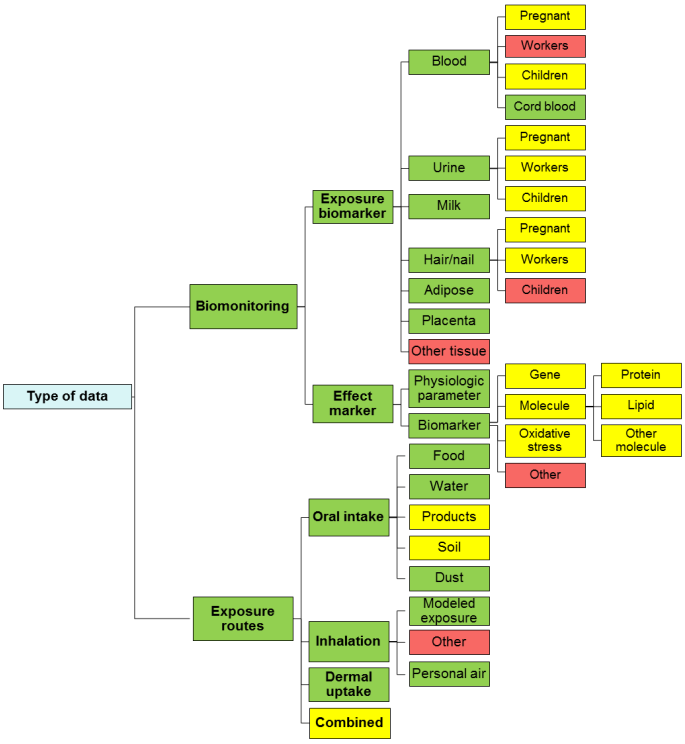


Figure 3.9: Illustration of the intrinsic evaluation performance for the exposure taxonomy. The colour coding is based on F_1-scores: Green ≥ 75%, yellow = 50–75%, red ≤ 50%.

of the 32 categories showing an improvement in F_1-score. When considering only the count of nodes that show an improvement in F_1-score accuracy, VC features are the least beneficial overall: 12 of the 32 categories improve in F_1-score when VC features are removed. On average, the analysis shows that all feature types benefit our classification methodology, as they all result in reduction of F_1-score when they are removed.

3.3.4 Case studies

Similar to our evaluation of the HoC, we conduct three case studies[25] that illustrate real-world applications of the Exposure Taxonomy and our classification methodology. We use exactly the same experimental setup described in Section 3.2.4.

Case study 1: Hexachlorobenzene *vs.* lead *vs.* 4-NP

In this case study, we compare hexachlorobenzene (HCB), lead, and 4-nonylphenol (4-NP). These three chemicals have distinctly different properties and exposure

[25] Case studies 1 and 2 were designed by two experts in Karolinska Institutet and implemented by the author. Case study 3 was designed and implemented by the author.

Table 3.14: Intrinsic evaluation results. All figures are expressed as percentages.

Class	Precision	Recall	Accuracy	F_1-score
Biomonitoring	94.9	95.5	93.1	95.2
Effect marker	89.0	81.8	90.7	85.3
Biomarker	89.4	69.2	94.2	78.0
Gene	92.6	61.7	98.3	74.0
Molecule	85.4	63.1	94.5	72.6
- Lipid	87.5	37.2	98.3	52.2
- Other molecule	80.0	42.9	96.9	55.8
- Protein	84.8	56.0	95.6	67.5
Other effect biomarker	91.7	33.8	98.8	49.4
Oxidative stress	82.1	51.6	99.0	63.4
Physiological parameter	84.0	70.1	90.8	76.4
Exposure biomarker	93.8	95.0	93.2	94.4
Adipose tissue	93.9	87.5	99.6	90.6
Blood	87.2	82.4	92.1	84.7
Hair/nail	97.7	89.9	98.4	93.6
Breast milk	91.6	86.4	99.0	89.0
Other tissue	86.0	25.9	96.9	39.8
Placenta	93.3	70.0	99.4	80.0
Urine	95.7	91.9	97.1	93.8
Exposure Routes	89.0	92.3	87.4	90.6
Combined	80.9	43.9	97.0	56.9
Dermal Exposure	83.9	72.6	98.0	77.8
Inhalation	92.4	81.5	93.3	86.6
Modeled exposure	91.1	83.1	98.0	86.9
Other inhalation	78.6	30.4	92.3	43.8
Personal air	90.5	74.8	97.9	81.9
Oral intake	87.7	82.1	88.1	84.8
Drinking water	83.4	81.3	95.7	82.3
Dust	90.4	75.1	97.4	82.0
Food	88.4	75.8	93.2	81.6
Products	80.4	42.3	96.4	55.4
Soil	78.5	62.5	97.7	69.6
Macro-average	88.0	68.4	95.6	75.5
Micro-average	90.3	81.9	95.6	85.9

routes, and therefore, our hypothesis is that the publication profiles predicted by our classifiers would reflect the expected exposure information associated with these chemicals.

HCB is a persistent organic pollutant which was used as a fungicide until the 1970s. Although it is now globally banned, HCB is still present in the environment and humans are exposed primarily *via* food (Gasull *et al.*, 2011; Newhook and Dormer, 1997). Figure 3.10 clearly shows that ingestion of food is the most studied exposure route, whereas other routes are poorly studied. HCB is a lipid-soluble compound predominantly measured in blood as well as in the body's fat-containing tissues, such as adipose tissues and mother's milk. This is reflected by the publication profile.

Table 3.15: Leave-one-out feature analysis results, all figures are F_1-scores. Values in parentheses indicate that leaving out that feature improved F_1-score.

Class	All	LBOW	GR	NE	VC	NB	MeSH	Chem
Biomonitoring	95.2	85.3	92.4	91.6	92.3	93.9	88.9	94.3
Effect Marker	85.3	83.2	84.6	82.9	83.1	83.1	79.6	80.1
Biomarker	78.0	72.3	(80.4)	73.9	76.9	73.2	70.0	71.7
Gene	74.0	65.6	73.9	70.0	71.5	72.9	69.0	72.3
Molecule	72.6	66.4	(76.7)	70.4	68.0	68.2	64.8	65.7
- Lipid	52.2	47.7	47.3	48.4	(53.2)	(53.4)	50.1	(53.3)
- Other molecule	55.8	46.6	53.6	55.1	45.4	47.5	49.4	46.0
- Protein	67.5	54.2	67.3	62.6	63.6	60.2	58.3	57.2
Other effect biomarker	49.4	(51.5)	45.3	46.4	(54.8)	(54.7)	(54.7)	(53.3)
Oxidative stress	63.4	62.7	62.7	59.2	(72.7)	(69.4)	(66.5)	(66.1)
Physiological parameter	76.4	70.2	75.3	(80.4)	70.5	71.2	69.6	68.6
Exposure biomarker	94.4	89.6	89.2	93.4	93.9	93.4	92.6	93.3
Adipose tissue	90.6	80.6	84.0	87.2	79.5	83.7	79.1	84.3
Blood	84.7	80.1	83.9	78.6	82.6	(85.2)	84.5	82.4
Hair/nail	93.6	84.2	91.6	91.7	(95.3)	91.9	92.3	(97.0)
Breast milk	89.0	72.0	85.4	88.9	83.1	78.6	76.8	76.2
Other tissue	39.8	25.6	36.2	36.9	28.5	28.4	27.9	26.6
Placenta	80.0	55.0	75.3	75.6	62.4	59.3	56.7	56.4
Urine	93.8	88.8	90.1	93.3	94.5	(96.0)	(95.4)	92.0
Exposure Routes	90.6	89.2	85.6	84.5	86.4	89.7	85.0	88.9
Combined	56.9	51.7	53.6	(57.2)	55.1	54.0	56.2	51.5
Dermal Exposure	77.8	71.2	71.9	(83.0)	74.9	74.8	73.6	71.4
Inhalation	86.6	78.9	78.0	(88.5)	85.2	85.7	86.5	83.2
Modeled exposure	86.9	83.8	80.5	84.6	(87.3)	83.9	78.9	80.5
Other inhalation	43.8	43.6	(47.0)	43.2	(45.6)	(44.4)	(45.1)	43.6
Personal air	81.9	77.8	74.8	(87.8)	(85.6)	81.5	76.8	(82.1)
Oral intake	84.8	76.2	78.7	81.1	(87.2)	82.8	81.4	82.0
Drinking water	82.3	73.8	76.1	80.9	(85.0)	80.9	80.2	78.7
Dust	82.0	80.1	80.2	77.3	(82.4)	80.8	79.0	79.7
Food	81.6	79.6	79.5	78.3	77.5	80.2	79.5	78.3
Products	55.4	50.4	54.2	54.8	(56.0)	52.7	52.2	51.6
Soil	69.6	58.0	68.8	(70.6)	62.2	65.2	61.0	64.8
Average	75.5	68.6	72.6	73.7	73.2	72.5	70.7	71.0

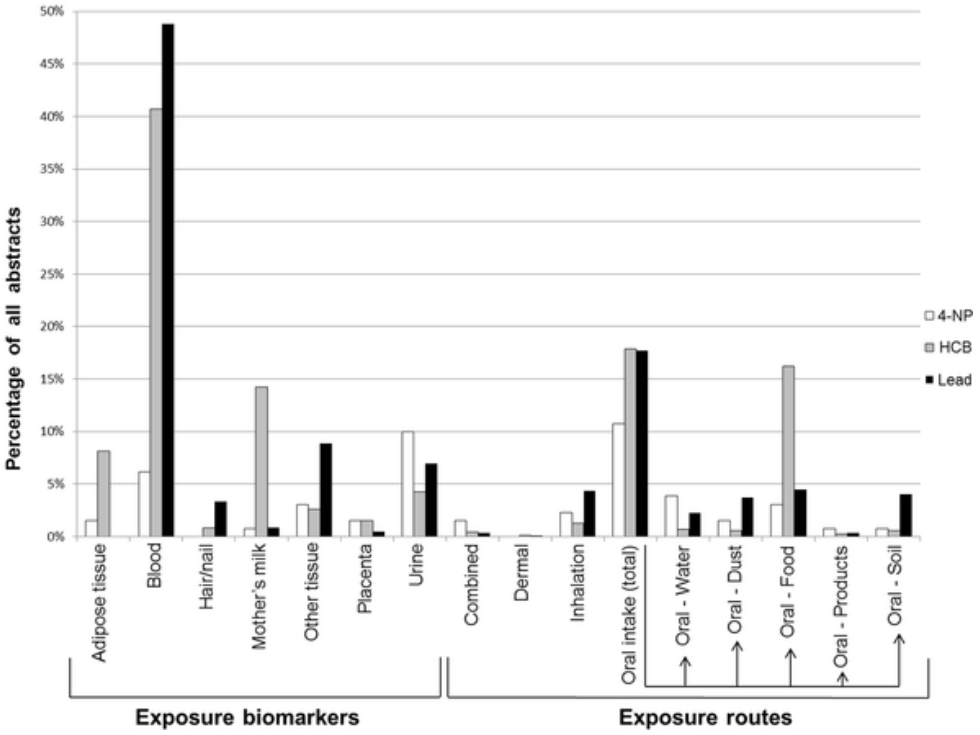


Figure 3.10: A comparison of exposure literature classification profile between 4-NP, HCB, and lead.

Lead is a naturally occurring toxic metal used in areas such as mining and smelting activities, and historically used in products such as paint and gasoline. Humans are mainly exposed to lead by ingestion of contaminated food or water and children can also be exposed by ingestion of dust and soil (CONTAM, 2009). For occupationally exposed individuals, inhalation is an important exposure route. According to Figure 3.10, ingestion of food, dust, and soil as well as inhalation are almost equally studied exposure routes. According to the publication profile, lead is predominantly measured in blood, followed by other tissues and urine, which is in accordance with current knowledge (Bergdahl and Skerfving, 2008). The high number of measurements in other tissues mostly considers lead in bones and teeth, which are established matrices for lead measurements.

A surprisingly high number of abstracts about lead and HCB were classified into the exposure biomarker sub-node "urine". This is because other chemicals have been measured in urine in the same articles in which HCB or lead have been measured in blood. Likewise, articles about HCB and lead have been classified into the exposure route sub-node *products* because another chemical (mercury) has been measured in amalgam fillings (considered as a product) while exposure to HCB or lead has been assessed from other exposure sources in the same abstracts. The classifier has not misclassified the abstract in these cases, however it was not able to

distinguish between different chemicals presented in the same abstract.

4-NP is an endocrine-disrupting chemical that has been used as a surfactant in industrial and household cleaning products and in the production of products such as paints and pesticides (Renner, 1997; Sharma *et al.*, 2009). Available human exposure information about 4-NP was found to be very scarce. Due to the low number of abstracts in each node for 4-NP, it is difficult to draw conclusions about the publication profile. However, Figure 3.10 shows that urine and blood are the most studied matrices for measuring exposure biomarkers, which is expected (Asimakopoulos *et al.*, 2012). The small number of abstracts allowed manual evaluation of the classification precision of all abstracts about 4-NP. With only a few exceptions, the abstracts were classified correctly (96% true positives, 4% false positives).

Case study 2: Differences in exposure data for phthalate esters

Phthalates are a group of industrial chemicals mainly used as plasticisers in PVC, but also in non-plastic products such as personal care products, paints and glues (Frederiksen *et al.*, 2007; Wittassek *et al.*, 2011). Humans are exposed primarily *via* the diet, but also from dust, air, and direct contact with consumer products (Janjua *et al.*, 2008; Wittassek and Angerer, 2008; Wormuth *et al.*, 2006).

This case study aims to map the distribution and amount of published exposure information available for six different phthalates: di(2-ethylhexyl)phthalate (DEHP), dibutylphthalate (DBP), butylbenzylphthalate (BBzP), diisobutylphthalate (DiBP), diisononylphthalate (DiNP), and diisodecylphthalate (DiDP). Due to their toxic effects, the use of DEHP, DBP, BBzP, and DiBP has been prohibited within the EU since 2015, whereas DiNP and DiDP are still in use and now substitute the banned phthalates. In this case study, we investigate whether the amount of available exposure data differs between the banned phthalates and those still in use.

All available abstracts in `PubMed` about these six phthalates were automatically classified to create publication distribution profiles. Figure 3.11 presents publication profiles for exposure biomarker measurements and exposure routes, which show that the levels of phthalates in humans are measured predominantly in urine as phthalate metabolites, whereas measurements in blood are less common (which is expected). Measurements in adipose tissue, hair/nail, and placenta are scarce or lacking for all phthalates. The publication profiles also reflect that we are mainly exposed to phthalates *via* food. Publication distribution profiles for effect biomarkers are presented in Figure 3.12.

This case study shows that the banned phthalates (DEHP, DBP, BBzP) are the most studied compounds, whereas there are considerable data gaps for the phthalates still in use (DiNP and DiDP). However, there is also little available data for the banned phthalate DiBP. The relatively larger amount of data for the banned compounds may be because these compounds have more established toxic properties and have historically been the most commonly used phthalates. However, DiNP and DiDP are among the most used phthalates in Europe today (KEMI, 2015; Wittassek *et al.*, 2011), which is a reason to request more exposure data for these

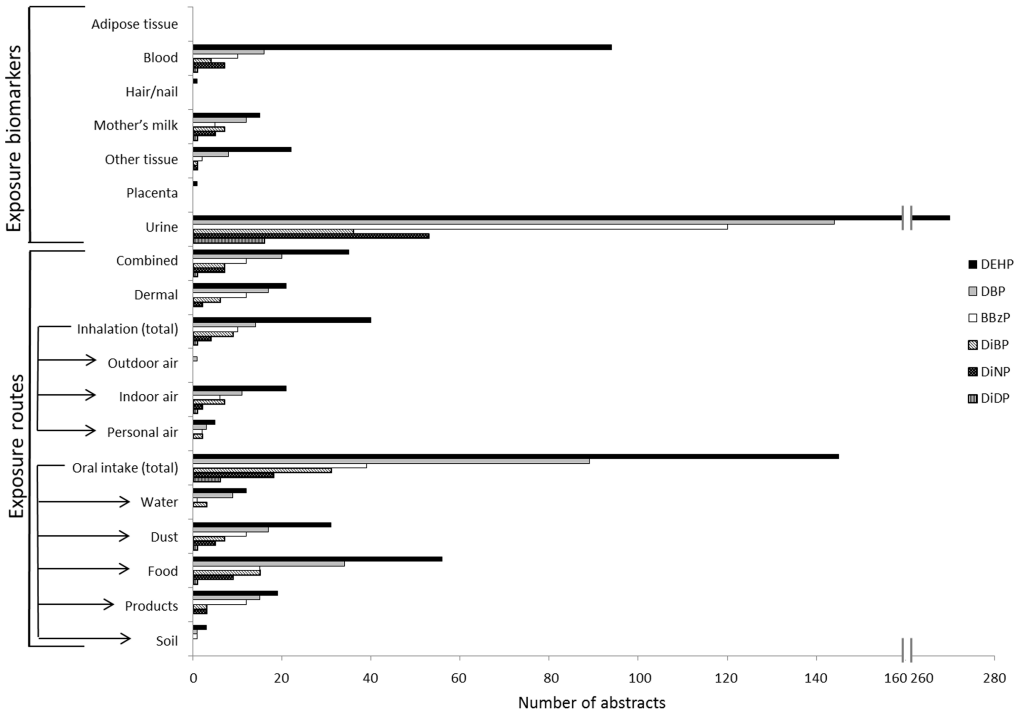


Figure 3.11: Publication profiles for exposure biomarkers and exposure routes for different phthalate esters.

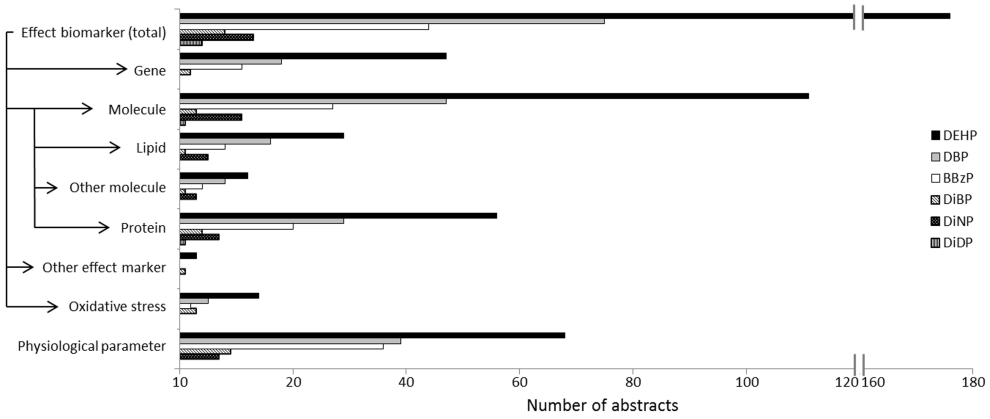


Figure 3.12: Publication profiles for effect biomarkers related to exposure to different phthalate esters.

Table 3.16: The number of abstracts retrieved by PubMed using keyword search query *vs.* classified by our system.

PubMed search query	Node in taxonomy	# of abstracts	
		Retrieved	Classified
7439-92-1 AND inhalation	Exposure routes – Inhalation	149	337
7439-92-1 AND (DNA OR gene) AND biomarker	Biomonitoring – Effect biomarker – Biomarker – Gene	65	120
7439-92-1 AND protein AND biomarker	Biomonitoring – Effect biomarker – Biomarker – Molecule – Protein	149	357
7439-92-1 AND blood AND biomarker	Biomonitoring – Exposure biomarker – Blood	407	3784
7439-92-1 AND (hair OR nail) AND biomarker	Biomonitoring – Exposure biomarker – Hair/nail	24	257

compounds. The overall pattern of phthalates may reflect general lack of exposure data for substituting compounds, whereas ample exposure data are available for compounds used for a long time. These findings raise concern that the research is not up-to-date with the current use of chemicals.

Case study 3: Information retrieval for Exposure science

In a similar manner to case studies 3 and 4 for the HoC evaluation (Section 3.2.4), we considered search queries that utilise our classification taxonomy. We selected lead (chemical abstracts service number **7439-92-1**) as the chemical of interest and we then retrieved abstracts from PubMed that are relevant to a certain type of exposure. Taking as an example inhalation, the search query would typically be formulated as "7439-92-1 AND inhalation" which retrieves abstracts from PubMed that should be relevant to both lead and inhalation. This particular search query yielded 149 results. However, when using our system in conjunction by classifying all lead-related documents according to the taxonomy and then observing how many were classified under the inhalation node in the taxonomy, we identified 337 abstracts—more than double the number identified by PubMed. This large difference exists because our NLP pipeline considers far more factors than just search term occurrences in the text. We ran five keyword search queries, along with our classifiers for the corresponding nodes in our taxonomy; the results of these are given in Table 3.16.

We can see in Table 3.16 that our method was able to identify far more abstracts than were retrieved by PubMed, in some cases by an order of magnitude. However, improving recall alone is not sufficient: we also need to at least keep the same level of precision achieved by PubMed. In order to approximate this, the top 20 abstracts retrieved from each PubMed search query were reviewed to ascertain whether each abstract is relevant for the given query. From this relevance judgement, we can approximate the precision and therefore percentage of false positive abstracts from

the PubMed search as well as our system's classification. Table 3.17 details the precision performance according to the top 20 retrieved abstracts.

In Table 3.17, we observe that our model's performance is substantially more precise and has far fewer false positives than the keyword based search in PubMed. We believe that in the cases where search queries have correspondence to our taxonomy, we can improve both precision and recall by a substantial margin because our classification system considers far more factors when judging for relevance according to the exposure domain than PubMed keyword searches. Therefore, our specialised system provides superior information retrieval for exposure science researchers.

Table 3.17: Manual evaluation of 20 abstracts retrieved from PubMed using **"lead"** search query *vs.* automatic classification into the corresponding node in our system. All figures are in percentages.

PubMed search query	PubMed keyword search		Our method	
	Precision	False Positive	Precision	False Positive
7439-92-1 AND inhalation	50	50	100	0
7439-92-1 AND (DNA OR gene) AND biomarker	35	65	100	0
7439-92-1 AND protein AND biomarker	35	65	100	0
7439-92-1 AND blood AND biomarker	85	15	95	5
7439-92-1 AND (hair OR nail) AND biomarker	60	40	75	20

3.3.5 Discussion

The intrinsic evaluation of our automatic classifier showed good performance as high F_1-scores were generally achieved. We have demonstrated that our classifier successfully categorises exposure data in a manner that greatly narrows down the amount of irrelevant information that would be retrieved by broad keyword based PubMed searches. We have also showed that our classifier is capable of finding more abstracts in specific sub-nodes and at the same time achieves higher accuracy compared to if the corresponding PubMed search strings were used. Consequently, the information that would be lost using specified PubMed searches is found by our classifier and the relevance of the retrieved information is higher.

The case studies showed that the classifier successfully can be used by researchers as a time-saving complement to manual information search. We also showed that the classifier creates chemical-specific publication profiles that reflect what we know about well-studied chemicals, which further confirms the relevance of the obtained results. As we show in our case studies, such profiles can be used to overview exposure data when comparing different chemicals or compounds within a chemical group and to detect data gaps.

The evaluation of the classifier revealed some challenges which should be addressed:

- The nodes other tissue for exposure biomarker measurements, other effect biomarkers and indoor air performed poorly, probably due to the diversity of data in these nodes. Using our gained knowledge about the characteristics of this diverse information, each node can be divided into more specific nodes, which provide more relevant classification and enhance the F_1-scores.
- The classifier is unable to distinguish between different chemicals that are measured in different tissues but presented in the same article. This is not a classification error *per se* since these abstracts are classified correctly based on the available information. However, chemical-specific publication profiles may show an incorrect pattern if the misclassification occurs systematically. Addressing this problem will require further analysis of the available annotations for textual cues that might help the classifier to make the distinction.
- Aiming to exclude animal and *in vitro* studies when retrieving abstracts from `PubMed`, only abstracts indexed with the MeSH term "humans" were used. However, some animal and *in vitro* studies were also labelled with this MeSH term and were therefore retrieved and included in the automatic classification. In these cases, our classifier classified the information correctly, but the interpretation of the results as exclusively human data may be misleading. We cannot change the manner in which abstracts are indexed by `PubMed` to overcome this problem. However, by extending the taxonomy to include animal and *in vitro* studies, we would overcome these biases.

3.4 Chapter Summary

Semantic text classification is the task of assigning one or more pre-defined labels to a span of text, where the label captures semantic information of interest.

We introduced two semantic text classification tasks: (i) the HoC, where the aim is to classify text according to a taxonomy of established traits and principles that describe the inception and proliferation of cancer, (ii) and the Exposure taxonomy, which classifies exposure assessment (a vital part of the chemical risk assessment).

We described a supervised NLP pipeline for classifying `PubMed` abstracts, and applied this pipeline for both of the described text classification tasks, and we evaluated our pipeline on both tasks intrinsically, and we found that we attain a good level of performance. We also conducted several cases studies that further support the usefulness of the classification by contrasting the results with real-world knowledge.

Chapter 4

Neural methods

In the previous chapter, we described semantic text classification and introduced two tasks that are relevant to the Cancer domain: the HoC taxonomy, and the Exposure taxonomy. We described a methodology that utilises a supervised NLP pipeline. Both datasets that were introduced were relatively small and sparse in nature (proportionally few labelled examples). This in part has led us to introduce a methodology that is heavy on feature engineering in order to mitigate the label sparsity in our data.

One of the main drawbacks of this approach is that handcrafting can be prohibitively time consuming, and heavily dependent on expert knowledge that is often difficult to elicit; since the resultant handcrafted features are domain dependent, it makes the trained classifiers difficult to port to other domains (Dai *et al.*, 2007; Sebastiani, 2002).

A possible option to these drawbacks is to use unsupervised and lightly-supervised methods, which bypass the need for labelled data. However, they tend to suffer from lower performance than supervised methods (Aggarwal and Zhai, 2012; Quan *et al.*, 2014; Zhang and Elhadad, 2013).

In this chapter, we explore the application of neural methods for semantic text classification. We consider document text classification (`PubMed` abstracts), as well as sentence classification. We will largely focus on the HoC task, but also investigate other semantic text classification tasks.

We will first describe a new method that jointly learns multi-level embeddings with class labels for semantic text classification. We will then delve into deeper neural architectures with the aim of automatically learning features using CNNs, and how these models can be better adapted to predict labels of a taxonomy in text.

4.1 Multi-level embeddings

We present a new approach to semantic text classification that is particularly beneficial for small labelled datasets. Our approach builds on the distributed memory (DM) model by Le and Mikolov (2014). The fast and simple unsupervised DM

model acquires paragraph level embeddings. We improve the model to jointly learn multi-level embeddings that encode class-label topical information in addition to text.

We jointly learn a model that captures embedding representation for the target class labels, as well as word-, sentence-, and document-level representations in the same space. From these multi-level embeddings we derive a set of features. Our approach requires no manual feature engineering, can cope with small amounts of labelled data and produces features that are more robust to domain variation, and is portable across domains.

At the document-level, the overall "topic" is a mixture of the sub-topics of paragraphs in that document. The topics of the paragraphs are in turn mixtures of the sentence topics, all the way down to word-level semantics. Our multi-level embeddings model captures this intuition elegantly. For example, an article about cars might have the first sentence discussing car manufacturing, followed by another discussing car safety, *etc.*. Each of these topics can be represented by sentence-level embeddings, while a document-level embedding can capture the overall topic of the article. We apply our approach at two different levels of granularity: document-level and at sentence-level. At the sentence-level, and we compare our approach under a supervised setting against a handcrafted method as well as semi-supervised classification using our multi-level embedding features.

4.1.1 Related work

Embedded distributed representations have been used widely for document and sentence classification. For example, Huang *et al.* (2014) learn document-level embeddings using word-level embeddings as input. Yan *et al.* (2015) learn document-embeddings by combing a Deep Boltzmann Machine and a Deep Belief Network. Bhatia *et al.* (2015) learn embeddings for large multi-label classification in situations where the label set is extremely large. Liu *et al.* (2015) use latent topic models to learn a topic from each word, and then learn an embedding based on both the topic and the word. Yogatama and Smith (2014) use structured regularisers based on parse trees, topics, and hierarchical word clusters, as well as hierarchical sparse coding for regularization using stochastic proximal methods (Yogatama *et al.*, 2015). All of these works have been trained and evaluated on general domains such as newswire rather than on sparse domains with small labelled datasets.

4.1.2 Methodology

This section first describes the DM model, and then explains how it was improved for sparse domain text classification by introducing jointly learned multi-level representations. We describe three types of features that we extract from such representations, and the fixed classification setup for our task-based evaluations.

The distributed memory model

The distributed memory (DM) model is an extension of the CBOW model of Mikolov *et al.* (2013b). The DM model learns a representation of a paragraph that captures the semantics of a paragraph's "topic". In the model, every word is represented in a word embedding matrix, and every paragraph in a paragraph embedding matrix. Paragraph representations are averaged or concatenated to predict the next word in a context using a hierarchical softmax classifier.

DM introduces an additional component to the model that allows a representation of the paragraph (*via* paragraph ID), which is treated internally like any other word in the model's vocabulary. It acts as a memory that remembers what is missing from the current context. The model learns a vector representation of the paragraph that captures its overall topic semantics *via* stochastic gradient decent.

Joint learning of multi-level embeddings

We improve DM by learning distributed representations that capture the topical information at varying levels of granularity—that is, we learn embeddings at a word-, sentence- (or paragraph-), and document-level. We also learn a distributed representation of the class labels, since these can be viewed as another level of abstraction that is more abstract than the document-level.

Our intuition is that jointly learning representations at different levels of granularity (including that of class label) provides us with better embeddings for text classification than learning a representation at each level separately. Each level captures different topic semantics, ranging from word-level to the class label. Figure 4.1 illustrates our model.

In Figure 4.1, words from word embedding matrix **W**, sentences from sentence embedding matrix **S**, documents from document embedding matrix **D**, and class-labels from class embedding matrix **C** are used as the context from which to predict the target word. That is, given a sequence of training words $w_1, w_2, w_3, \ldots, w_T$ that belong to sentence s_t in document d_t, which has also a set of associated classification labels $c_1, \ldots, c_m$, the objective of the model is to maximise the average log probability:

$$\frac{1}{T}\sum_{t=k}^{T-k} \log\ P(w_t \mid w_{t-k}, ...w_{t+k}, s_t, d_t, c_1, ...c_m) \tag{4.1}$$

We use a softmax output layer to obtain the probability of the target word given its context:

$$P(w_t \mid w_{t-k}, \ldots, w_{t+k}, s_t, d_t, c_1, \ldots, c_m) = \frac{e^{\vec{y}_{w_t}}}{\sum_i e^{\vec{y}_i}} \tag{4.2}$$

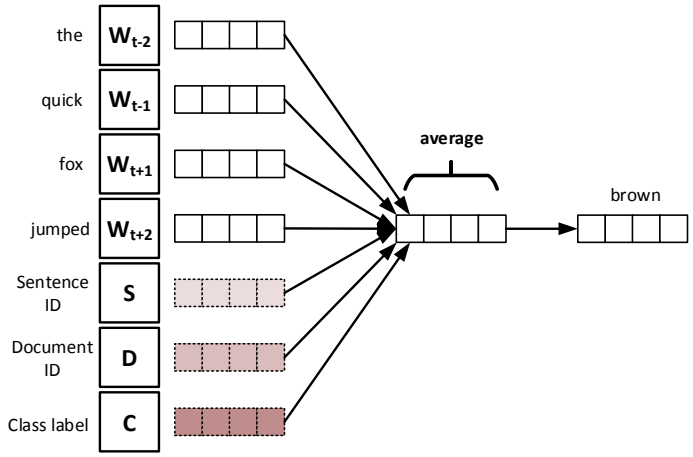


Figure 4.1: Illustration of distributed joint learning of different granularities of text contexts: words (**W**), sentences (**S**), documents (**D**), and classes (**C**). The model predicts the target word (w_t) based on the semantics captured by all these contexts. Shades represent level of abstraction/granularity.

where each y_{w_t} is calculated as:

$$\vec{y}_{w_t} = \mathbf{U}\frac{\sum_{i=-k}^{k} \vec{w}_{t+i} + \vec{s}_t + \vec{d}_t + \sum_{i=1}^{m} \vec{c}_i}{2k + m + 2} + b$$

where $k \neq 0$, **U** is the weight matrix, b is the bias, and we average the word vectors extracted from **W**, the sentence vectors extracted from **S**, similarly, the document vectors from **D**, and class label vectors from **C**.

Extracting features

We extract three types of features from the jointly-learned multi-level representations: the sentence (or document) embeddings (EMBED), the distances between word embeddings (WORD-DIST), and the similarities between classes (CLASS-SIM).

- **Embedding features**: since embeddings are learned at different levels, we use the document-level embeddings when classifying at the document-level. Likewise for sentence-level classification, we use only the sentence-level embeddings. Word-level embeddings are only used to extract distance features.
- **Word distance features**: we measure the cosine similarity between each unique non-stop word embedding occurring in the input sentence or document with the embedding representation for a given class label, *i.e.*,

$$\delta_{\vec{w}_i}^{\vec{c}_i} = \cos(\vec{w}_i, \vec{c}_i) \tag{4.3}$$

where $\vec{w}_i$ is embedding for word w_i in the input text, and $\vec{c}_i$ is the embedding representation of a class label that has been jointly learned from the training data. The input text has variable length, so we represent these distances in sparse vector format using a dictionary of all non-stop words in the corpus labelled with the given class c_i—a "bag of word distances".

- **Class-similarity features**: Word distance measures capture the similarity between words and class labels, but not between phrases or sentences. For this, we use word-level embeddings to measure the semantic similarity between a class and target text (sentence or document) using the earth mover's distance (EMD)[26], or the energy distance of moving a distribution.

EMD has been used successfully in image retrieval (Rubner *et al.*, 2000), document topic similarity (Wan, 2007), and more recently in combination with word embeddings (Kusner *et al.*, 2015). This method is useful for estimating the similarity between text with varying word count and overlap: the sentence *"sipping a cup of tea"*, for example, should have a relatively small EMD compared to *"wine tasting"*, despite them having no overlap and being of different length. Kusner *et al.* (2015) formulate the EMD problem as a linear program that can be expressed as the following optimisation:

$$\text{emd}\,(d, d') = \min_{\mathbf{T} \geq 0} \sum_{i,j=1}^{n} \mathbf{T}_{ij} ||\vec{x}_i - \vec{x}_j||_2 \tag{4.4}$$

subject to the following flow constraints:

$$\sum_{j=1}^{n} \mathbf{T}_{ij} = \vec{d}_i \quad \text{and} \quad \sum_{i=1}^{n} \mathbf{T}_{ij} = \vec{d'}_j \tag{4.5}$$

Here, $\mathbf{T} \in \mathbb{R}^{n \times n}$ is a flow matrix, *i.e.*, $\mathbf{T}_{ij}$ denotes how much of word i in the source document d travels to word j in the destination document d', and $\vec{x}_i$, $\vec{x}_j$ are embeddings for words i and j. Class-similarity features are obtained by finding the minimal distance between a given input (either document or sentence) and the given class, where only the most discriminatory word embeddings for the given class are combined, *i.e.*, non-discriminatory words that occur in all classes are excluded[27]. We then use Equation 4.4 to measure the similarity between words occurring in the text and the class combined word list.

[26] Also known as the Wasserstein metric.

[27] We discarded all words in the training set that occur in more than 80% of all class contexts as non-discriminatory.

Supervised classification

We apply a fixed classification setup in order to compare our new method against several embedding baselines as well as handcrafted classification. Similar to the experimentation setup described in Section 3.2.3, we use SVMs with a linear kernel, implemented using scikit-learn (Pedregosa *et al.*, 2011), and perform a standard grid search for kernel regularisation parameter selection.

We use L_1 and L_2 normalisation of input features, weighted equally according to the three aforementioned types (embedding, class-similarity and word-distance), *i.e.* features within each type are normalised separately and then combined.

We perform a 4-fold cross-validation setup and 5-fold nested cross-validation for kernel parameter tuning (using grid search)—*i.e.* we do a 5-fold cross-validation grid search nested in each of the outer four folds.

Semi-supervised classification

In a semi-supervised setting, we use vast amounts of unlabelled data, *i.e.* documents/sentences unlabelled with any class information, and a much smaller amount of labelled documents/sentences. Instead of using a supervised classifier to learn the decision boundaries, we use the distance measurements and a tuned cut-off threshold for each class to determine class assignment.

We use WORD-DIST and CLASS-SIM (Section 4.1.2), and EMB-DIST: the cosine distance between the embedding of a sentence or document and an embedding of a class label. A cut-off threshold is used to determine positive or negative classification for each class. Under the WORD-DIST setup, we average all of the word distances. We perform a grid search for this threshold on 10% held-out data.

4.1.3 Evaluation

We apply our methodology to two real-life biomedical text classification tasks: Hallmarks of Cancer (HoC) classification, and rhetorical text classification. We evaluate these tasks on document- and sentence-level classification.

We first introduce rhetorical text classification. Also known as information structure analysis, rhetorical text classification segments scientific text into information categories. One such classification technique is argumentative zoning (AZ) (Teufel and Moens, 2002), which captures the rhetorical progression of the scientific argument by segmenting a document into several zones, such as "Objective", "Background", "Method", "Result", and "Conclusion".

This task differs from the HoC in that the objective is to classify scientific text according to generic labels (*i.e.* unrelated to domain-specific knowledge) and the focus is on different classification features, such as the position of the text in the document and the author's writing style. For example, the "Objective" zone of the argument generally appears very early in the article using an active voice.

Table 4.1: Description of argumentative zones and their distribution in the annotated data.

Class	Description	# Abstracts	# Sentences
Objective (*OBJ*)	The background and the aim of the research	744	812
Background (*BKG*)	The circumstances pertaining to the current work	692	1517
Method (*METH*)	The way to achieve the goal	640	1617
Result (*RES*)	The principal findings	889	4028
Conclusion (*CON*)	Analysis, discussion and the main conclusions	859	1484

We evaluate the HoC task using our corpus described in Section 3.2.1, and for Rhetorical text classification, we evaluate using an expert-annotated dataset from Guo *et al.* (2010) comprising 1000 PubMed abstracts relevant to cancer biology.

The dataset consists of 7985 labelled sentences, with an inter-annotator agreement of $\kappa = 0.85$. There are five mutually non-exclusive classes, described together with their frequencies in Table 4.1.

We compare the multi-level embeddings method with the fully supervised hand-crafted NLP pipeline describe in Section 3.2.2

Many of the features used for this task are similar to those used in the HoC classification task, namely BOW, noun bigram, and GR. Here we also include POS tags, and the following task-specific features:

Location: categories tend to appear in typical positions in a document. For example, *BKG* usually occurs at the beginning and *CON* at the end. The abstract is divided into ten equal parts and the location of a sentence is defined by the parts where the sentence begins and ends.

History: the category of the preceding sentence is used as a feature. This is because certain categories tend to appear before others. For example, *RES* tends to be followed by *CON* rather than other categories.

Voice: there is a correlation in scientific writing between the active and passive voice and certain categories, for example, passive voice is more frequent in *METH*.

In addition to the handcrafted model, We also include the following baselines:

- We train SGNS representations on the corpus, and obtain sentence or document-level embedding using a composition function $f(w_i, \ldots, w_n)$, where f is either addition (ADD), averaging (AVG), or the maximum (MAX). We do the same with CBOW representations. The resultant composed embeddings are used as input features for the classifier. For conciseness, we include only the best performing composite function here.
- We compare against standard BOW classification, where each non-stop word in the corpus is a binary feature.

4.1.4 Results

In this section, we present the results of our experiments and compare our method to the handcrafted models in addition to several baselines detailed below.

The first experiment is to classify the HoC text into documents and sentences. Table 4.2 shows the results for both. We can see that sentence level classification is more difficult, due to the smaller amount of context information available. The table also shows the results for the composed embedding baselines, the supervised BOW baseline, and the handcrafted supervised model. This is followed by the three feature types (EMBED, CLASS-SIM, WORD-DIST) in all possible combinations, and finally the full model using all features.

With regards to the EMBED feature type, we distinguish between learning the representation independently (*e.g.* embeddings are learned without knowledge of the document) or jointly as described in Figure 4.1. We can see that the EMBED features by themselves perform better than any of the embedding baseline models. Jointly learning embeddings improves the F_1-score by approximately 4-5% for both document and sentence classification.

Table 4.2: HoC multi-level embeddings experiment results. All figures are micro-averages expressed as percentages.

	Document classification			Sentence classification		
Model	Precision	Recall	F_1-score	Precision	Recall	F_1-score
SGNS	43.7	25.9	32.5	13.6	23.3	17.2
CBOW	30.9	27.6	29.2	7.5	15.1	15.0
BOW	47.9	37.9	42.3	53.8	24.1	33.3
HANDCRAFTED	82.8	69.4	75.5	59.2	46.4	51.4
EMB-DIST SEMI-SUPERVISED	24.3	28.5	26.3	25.4	25.4	21.9
WORD-DIST SEMI-SUPERVISED	30.9	36.5	33.5	43.7	24.6	31.5
CLASS-SIM SEMI-SUPERVISED	**44.1**	**38.8**	**41.3**	**26.7**	**42.1**	**32.6**
EMBED-IND	44.0	37.4	40.4	26.5	48.0	34.2
EMBED-JOINT	54.2	46.4	49.9	37.6	39.6	38.6
CLASS-SIM	80.2	49.9	59.4	36.2	45.8	40.5
WORD-DIST	58.5	51.9	55.0	32.7	40.3	36.1
EMBED + CLASS-SIM	69.3	58.3	63.3	54.7	52.1	53.3
EMBED + WORD-DIST	60.9	60.4	60.7	54.6	56.3	55.4
CLASS-SIM + WORD-DIST	64.5	72.7	68.4	61.5	61.0	61.2
EMBED + CLASS-SIM + WORD-DIST	**85.5**	**69.8**	**76.4**	**77.7**	**60.1**	**67.6**

When considering the three features types, CLASS-SIM outperforms both EMBED and WORD-DIST, with a particularly notable improvement in document classification.

When pairing the three feature types, the combination CLASS-SIM + WORD-DIST gives the best results, as is consistent with their individual results. Finally, when combining all three features, the full model outperforms all baselines by a significant margin, especially notable for sentence-level classification. Regarding the semi-

supervised models, using class similarity CLASS-SIM alone significantly outperforms using word cosine distance WORD-DIST, and document-embedding to class-embedding distance EMB-DIST.

We also investigate the performance of our model and baselines when subjected to domain variation—when we learn the embeddings from a different domain than that of the classification task. We experimented by learning the embeddings using the Wikipedia corpus. We seed the model with the labelled HoC training data and then train on Wikipedia instead of domain specific literature acquired from `PubMed`.

Naturally, we expect all of the models to perform worse with Wikipedia-trained embeddings than with domain specific embeddings. This is indeed what happens (Table 4.3). However, some models prove more robust than others—their drop in F_1-score accuracy is smaller. By this measure, our full model and the semi-supervised models are less susceptible to domain variation with both document and sentence-level classification.

Table 4.3: Document and sentence classification micro-averaged F_1-score (%) using domain-specific and Wikipedia embeddings.

	Document		Sentence	
Model	Domain	Wikipedia	Domain	Wikipedia
SGNS	32.5	18.4	17.2	11.1
CBOW	29.2	14.3	15.0	10.4
SEMI-SUPERVISED [28]	41.3	35.3	32.6	28.6
FULL MODEL	**76.4**	**61.5**	**67.6**	**54.6**

Rhetorical text classification results

Our objective is to classify scientific text according to five argumentative zonings. Table 4.4 summarises the results. Similar to HoC classification results, all three feature types perform significantly better than the embedding baseline models. When analysing the three feature types separately, WORD-DIST outperforms the other two. EMBED + WORD-DIST is the best-performing feature pair.

The full model significantly outperforms all baselines. However, it does not match the handcrafted approach. This is because the most influential feature in this task is the location of the text (Guo *et al.*, 2011; Kiela *et al.*, 2015). As our model does not take any word or sentence ordering into account, it would be difficult to compensate for the location feature. If, however, we include the location feature in addition to the three feature types in the SVM classification, our model outperforms the handcrafted baseline by a 1.4% difference. Admittedly, this would make the model slightly handcrafted by itself, but no additional work is necessary to get this feature and it does not vary across tasks or domains. This shows that, for our model, including location information provides better features for this task

[28] Semi-supervised model uses CLASS-SIM.

than the handcrafted approach including location information. Looking at the semi-supervised models, the results suggest that CLASS-SIM outperforms the other feature types by an even larger margin than for Evaluation Task 1.

Table 4.4: Rhetorical text classification Micro-averaged performance comparison. All figures are percentages.

Model	Precision	Recall	F_1-score
SGNS	45.6	31.5	37.3
CBOW	47.3	30.9	37.4
BOW	54.8	35.1	42.7
CNN	74.9	66.9	70.7
HANDCRAFTED	88.9	85.0	86.9
EMB-DIST SEMI-SUPERVISED	23.7	38.9	29.4
WORD-DIST SEMI-SUPERVISED	36.6	28.8	32.2
CLASS-SIM SEMI-SUPERVISED	**57.1**	**40.9**	**47.7**
EMBED IND	43.3	41.9	42.6
EMBED JOINT	57.6	37.7	45.6
CLASS-SIM	58.7	46.0	51.6
WORD-DIST	55.2	51.8	53.5
EMBED + CLASS-SIM	64.5	59.9	62.1
EMBED + WORD-DIST	78.1	57.5	66.3
CLASS-SIM + WORD-DIST	82.0	54.5	65.5
EMBED + CLASS-SIM + WORD-DIST	**81.2**	**72.7**	**76.7**
FULL MODEL + LOCATION	**89.6**	**86.9**	**88.3**

4.1.5 Discussion

The intuition behind multi-level embeddings is that each level captures slightly different topical semantics. We therefore employ these embeddings to produce three types of features that require no additional data or labour, that are efficient to extract and much easier to port than handcrafted features. We have shown how these feature types can be used with standard classification algorithms such as SVMs and with semi-supervised classification where the decision boundaries are not learned from labelled data.

In the first task (HoC) our approach matched or outperformed a handcrafted fully-supervised approach. The model performed substantially better at sentence-level classification which had much less context than the document-level classification. We also showed that our features are less susceptible to domain variation. In the second task (rhetorical text classification), the full model outperformed all baselines, as well as the handcrafted approach when including location information in the classification process. We found that even when using only the semi-supervised setup, we can outperform fully supervised models that are trained with embeddings obtained from the baseline embedding algorithms (SGNS and CBOW). Despite this, this approach does have some drawbacks:

- Although this approach reduced the dependency on heavy feature engineering, it still involved some feature engineering, *i.e.* the construction of features in addition to the embeddings (WORD-DIST and CLASS-SIM).
- The extraction of the CLASS-SIM feature using EMD, can be slow with large volume of text. It requires an optimisation which is solved in $O(n^3 \log n)$, where n is the number of unique words in document. However, Rubner *et al.* (2000) reduce this to linear time by assuming some constraints.
- This approach does not take word order into account, as is the case for the DM model. This is problematic in tasks where word order is important, but can also affect the semantic text classification task—for example, the effect of word ordering on negation.

4.2 Convolutional neural networks

A CNN is a type of forward-feed neural network that has two key advantages over other forward-feed networks such as multilayer perceptrons (MLPs): it automatically learns features from the input data, and it greatly reduces training time as it has fewer parameters to learn compared to large fully-connected MLPs. CNN architectures can generally be summarised by four key components:

- **Convolution**: the process of automatically extracting features from the input, this is done by learning a feature maps (also known as an activation map or filter). A feature map is initialised with random values, but with training, the model learns useful distinctive features relevant to the given classification task. A convolution is simply computing the dot product between feature map and the network inputs (embeddings).
- **Subsampling (pooling)**: reduces the dimensionality of the feature maps but retains the most important information. The subsampling operator can be of different types, *e.g.* max, min, sum, mean, *etc.*Max-pooling is the most commonly used subsampling operator, and is known to work best. A moving window of size n is moved at step size t. The largest element from the window is recorded in the output subsample, and the process is repeated at each step t.
- **Non-linear activation**: a non-linear activation layer (typically implemented using a rectified linear unit (ReLU)) enables the network to capture non-linear combinations of the input signal (as both the convolution and subsampling operators are linear operators).
- **Classification**: the final layers in the network are fully connected dense layers—in essence, an MLP with the number of hidden layers and units as hyperparameters to the model.

In the following sections we describe an implementation of a CNN for classifying the Hallmarks of Cancer (HoC). We will first briefly describe related work, then our methodology and evaluation.

4.2.1 Related work

CNNs were first proposed for image processing (LeCun and Bengio, 1995) and have been recently shown to achieve state-of-the-art performance in a range of NLP tasks, in particular in text classification (Severyn and Moschitti, 2015; Zhang *et al.*, 2015; Zhang and Wallace, 2015). While neural network-based methods in general and "deep" networks in particular are increasingly popular for general domain NLP, there has been comparatively little work applying this class of methods to biomedical text. One recent study applying a CNN model to biomedical text classification task was presented by Limsopatham and Collier (2016), who applied CNNs to the task of adverse drug reaction detection in social media messages (Ginn *et al.*, 2014). In addition to the specific subdomain of the source texts and the novel categories represented by the HoC, one factor that sets apart the task here from this previous work is the length of the texts: instead of sentences or brief social media messages, our task involves the classification of publication abstracts typically consisting of hundreds of words.

4.2.2 Methodology

We base our CNN architecture on the simple model of Kim (2014) which consists of an initial embedding layer that maps input texts into matrices, followed by convolutions of different filter sizes and 1-max pooling, and finally a fully connected layer. The architecture is illustrated in Figure 4.2.

We implemented the neural network using Keras (Chollet, 2015). Model hyperparameters and the training setup were initially based on those applied by Kim (2014), summarised in Table 4.5.

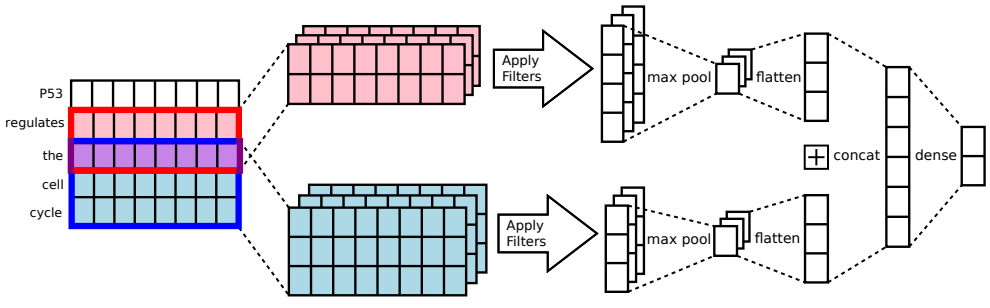


Figure 4.2: CNN network architecture.

Table 4.5: Kim (2014) model parameters.

Parameter	Value
Word vector size	300 (Google News vectors)
Filter sizes	3, 4, and 5
Number of filters	300 (100 of each size)
Dropout probability	0.5
Minibatch size	50

Some of these parameters were further refined in experiments using only the training and development portions of the data (Section 4.2.4). In the final test set experiments, we evaluate the network using both the set of parameters used by Kim (2014) as well as with those selected in our development set experiments. We train the models for 20 epochs using categorical cross-entropy loss and the Adam optimization method (Kingma and Ba, 2014).

For regularisation, we only apply dropout (Srivastava *et al.*, 2014) before the output layer. We also considered L_2 regularisation but did not find a consistent improvement in preliminary experiments.

Word embeddings

The first layer of the CNN involves mapping words in the input to dense, low-dimensional vectors. These word embeddings are critically important as they represent the "meaning" of the words in the model, *e.g.* how similar one word is to another. Although it is possible to learn these embeddings from scratch (random initialisation) during the normal training process, recent studies have shown that it is effective to use embeddings that have been separately induced on large, unannotated corpora (Collobert *et al.*, 2011; Kim, 2014). Work in BioNLP has further established that word embeddings are domain-dependent: to get the maximal benefit from using pre-trained embeddings for biomedical NLP tasks, the embeddings must be induced using biomedical texts (Stenetorp *et al.*, 2012).

Table 4.6: Word embedding initialisation for multilevel embedding.

	Source texts		Vectors			
Name	domain	size	words	dim	OOV	Reference
Google News	General	100B	3M	300	31.0%	Mikolov *et al.* (2013b)
Pyysalo PM	Bio	3B	2.3M	200	0.52%	Pyysalo *et al.* (2013a)
Pyysalo PMC	Bio	2.5B	2.5M	200	0.51%	Pyysalo *et al.* (2013a)
Pyysalo PM+PMC	Bio	5.5B	4M	200	0.49%	Pyysalo *et al.* (2013a)
Pyysalo Wiki+PM+PMC	Both	7.5B	5.4M	200	0.53%	Pyysalo *et al.* (2013a)
Chiu win-2	Bio	2.7B	2.2M	200	0.49%	Chiu *et al.* (2016)
Chiu win-30	Bio	2.7B	2.2M	200	0.49%	Chiu *et al.* (2016)

We consider a variety of word embeddings induced using models implemented

Table 4.7: Annotation statistics.

	Train		Devel		Test		Total	
Hallmark	pos	neg	pos	neg	pos	neg	pos	neg
Sustaining proliferative signalling	328	975	43	140	91	275	462	1390
Evading growth suppressors	172	1131	22	161	46	320	240	1612
Resisting cell death	303	1000	42	141	84	282	429	1423
Enabling replicative immortality	81	1222	11	172	23	343	115	1737
Inducing angiogenesis	99	1204	13	170	31	335	143	1709
Activating invasion and metastasis	208	1095	29	154	54	312	291	1561
Genomic instability and mutation	227	1076	38	145	68	298	333	1519
Tumour promoting inflammation	169	1134	24	159	47	319	240	1612
Cellular energetics	74	1229	10	173	21	345	105	1747
Avoiding immune destruction	77	1226	10	173	21	345	108	1744

in the popular `word2vec` package (Mikolov *et al.*, 2013a). First, we use the general-domain Google News vectors also applied by Kim (2014).[29] Second, we evaluate three sets of word vectors induced on various combinations of `PubMed` (PM), PMC and Wikipedia texts by Pyysalo *et al.* (2013a).[30] Finally, we consider two variants of `PubMed`-based vectors introduced by Chiu *et al.* (2016).[31] The properties of these word vectors are detailed in Table 4.6. Note that unlike the other properties, the out of vocabulary (OOV) rate is not a characteristic of the word vectors alone, but the ratio of words in the task training data that do not appear in the work vectors. The high OOV rate for the Google News vectors is due primarily to removal of stopwords, punctuation, and numbers.

4.2.3 Evaluation

We divide the dataset into ten binary-labelled datasets (one per hallmark), where the positive examples in each are the abstracts annotated with the hallmark, and negative examples are those that are not. We split the annotated data into training, development and test subsets instead of applying the cross-validation setup used previously (Section 4.1.3). We do this because cross-validation setups using all available data fail to make a clear separation between data used for method development and blind data held out for final testing only, and should be avoided in studies involving experimentally driven model refinement (as we do here). Consequently, we initially split the corpus in 70/10/20% proportion to train, development and test sets with a random sampling strategy that aimed to roughly preserve the overall class distribution in each sample. The test set was held out during development and only used in the final experiments. Table 4.7 shows the distribution of positive and negative labels for each hallmark.

[29] Available from https://code.google.com/archive/p/word2vec/
[30] Available from http://bio.nlplab.org/
[31] Available from https://github.com/cambridgeltl/BioNLP-2016

We implement and evaluate two SVM-based methods and two CNN variants, described in the following. All of these machine learning methods are applied to the multi-label task by training ten binary classifiers, one for each hallmark label.

We implement a simple classifier using only BOW features as a basic SVM baseline. In the BOW approach each document is represented by the set of words appearing in it, discarding word order and frequency information. For training the model, we use the linear kernel SVM implemented in the Scikit-learn (Pedregosa *et al.*, 2011) toolkit. We fine-tune the regularisation hyperparameter C conventionally using evaluation on the development dataset with a search between 10^{-2} and 10^{2} on a log scale.

We also use the NLP pipeline (Section 3.2.2). All features are extracted from the training dataset and are then filtered by frequency to remove features that are too common or too rare, leaving behind only the most discriminating features. We use a linear kernel and fine-tune the regularisation parameter C on the development dataset using the same process applied for the BOW model.

We train 10 binary classifiers one for each of the hallmarks in a one-*vs*.-rest (OVR) setup. As there are significantly more negatively labelled documents than positives, we use inverse class weighting in order to correct for the class imbalance when training the classifiers.

Classifier performance is evaluated using the standard precision, recall, and F_1-score metrics as well as with the AUC, which is more sensitive in summarising performance over all possible classification thresholds and eliminates the need to pick a specific threshold for evaluation.

As the dataset is comparatively small and the number of positive examples in particular is very limited for many labels, the random factors in CNN initialisation and training can have a substantial effect on the resulting model. To address this issue, we systematically repeated each CNN experiment 10 times and report the mean of the evaluation results.[32]

To help mitigate overfitting in the CNN, we apply early stopping by testing only the model that achieved the highest results on the development set, in addition to other measures used in the model (such as dropout).

4.2.4 Results

We summarise the results of the CNN evaluation to the task using the development data, and then present the comparative results on the test set.

Development results

We considered a range of modifications to the basic CNN model to better adapt it to biomedical domain text classification in general and the specific task studied in this work in particular. Of these modifications, evaluation on the development set identified three that appeared to have beneficial effects on performance: oversampling

[32] As SVM optimisation is convex, repetitions are unnecessary.

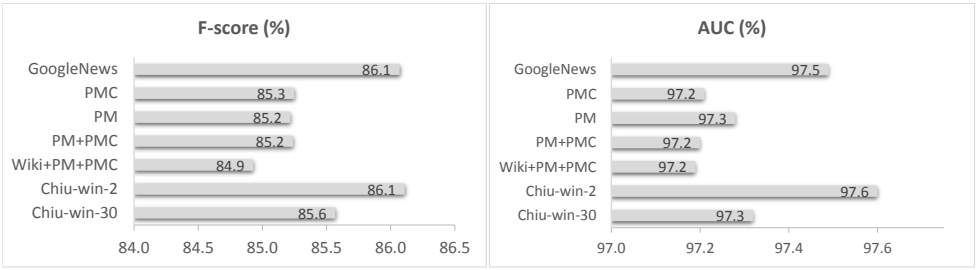


Figure 4.3: Embedding performance (macro-averaged) on the development dataset

to address the class imbalance, using in-domain word vectors, and adjusting the filter sizes to the task. We next briefly describe these modifications and the associated results.

Table 4.8: Average development set performance with and without oversampling

	F_1-score	AUC
Standard	82.3%	97.3%
Oversampling	85.0%	97.5%

The dataset is highly biased, with negative examples outnumbering positives more than 10-fold for a number of the labels (Table 4.7). Standard training on such data is likely to result in models with high precision, low recall, and thus comparatively low F_1-scores. Addressing this, we oversample the positive examples in the training set with replacement so that their number matches that of the negatives. This modification increased the average F_1-score on the development set from 85.3% to 86.1%. As expected, the effect on the distribution-independent AUC metric was more limited, improving from 97.3% to 97.5% with oversampling.

As discussed in Section 4.2.2, the word vectors used to initialise the embedding layer of the network can have a significant effect on performance. We trained the models using each of the word vectors shown in Table 4.6 with oversampling (see above) and evaluated development set performance using the maximum F_1-score and AUC metrics. The results are summarised in Figure 4.3.

Surprisingly, we find that the general domain Google News vectors give very competitive performance despite their high OOV rate (see Table 4.6), outperforming all in-domain vectors with the exception of the window size 2 word vectors of Chiu *et al.* (2016). Even these biomedical word vectors only show very modest advantage over the Google News vectors for AUC. In the last development set experiments below and the final test set experiments, we apply the `PubMed`-based vectors induced with window size 2 from Chiu *et al.* (2016). that were shown to give the best results here.

We experiment with varying the number of filter sizes in the convolutions. The base model of Kim (2014) uses three filter sizes (3, 4, 5); as part of our hyperparameter

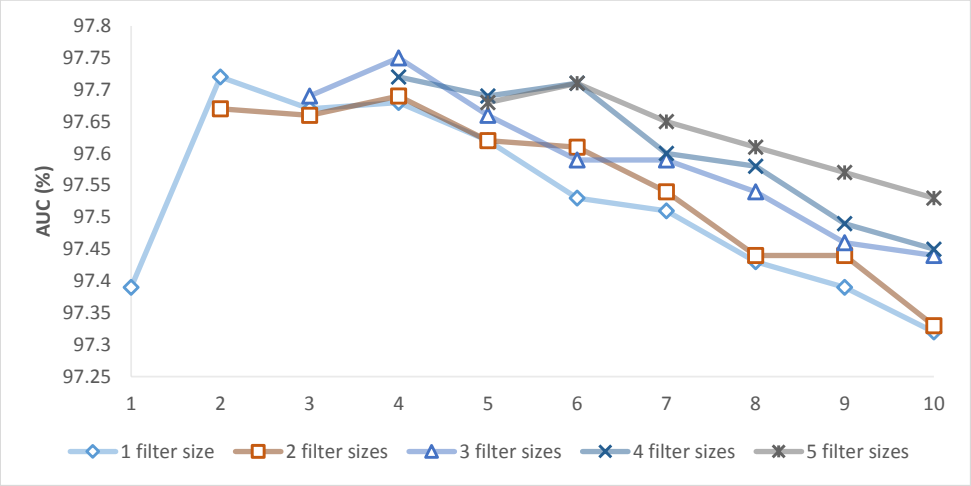


Figure 4.4: Macro-average AUC with respect to a varying number of filter sizes. Each point on the graph represents the maximum size of filter used (*e.g.* for *2 filter sizes*, performance with filters of sizes 2 and 3 is plotted at 3).

search, we investigated what happens to the performance (AUC) with respect to varying filter sizes (1–10) and numbers of filter sizes (1–5), while keeping the total number of filters constant at 300 and filter sizes are ordered consecutively. Figure 4.4 shows that performance generally falls when increasing the filter size, and the best performance is achieved using three filters of sizes (2, 3, 4). Another important observation is that the variation in performance is not very substantial, implying that the model is fairly robust to the specific setting of this parameter.

Test results

The results of the evaluation on the test data are shown in Table 4.9 for F_1-score and 4.10 for AUC. Overall, both metrics agree that the SVM with bag-of-words features has the lowest performance, and the CNN tuned to the task the highest. As could be expected, the SVM with rich features outperforms the base CNN in terms of F_1-score; however, the latter, generic model achieves a notably higher AUC than the SVM, suggesting that the slight advantage of the former for F_1-score may be due in part to a better position of the decision boundary.

The CNN tuned to the task achieves the highest performance on average by both metrics, and further has the highest performance for 7/10 individual classification tasks in terms of both F_1-score and AUC, outperforming the previous state-of-the-art on this dataset.

The CNN-based classifiers fall notably behind the SVM-based ones for the *Cellular energetics* hallmark, where even the BOW-based SVM clearly outperforms the tuned CNN for both metrics. A similar, but less pronounced advantage in favor of the SVM is seen for the *Inducing angiogenesis* hallmark.

Table 4.9: Comparison of test results using F_1-score

	SVM		CNN	
Hallmark	BoW	Rich	Base	Tuned
Sustaining proliferative signalling	70.0%	67.4%	66.3%	67.9%
Evading growth suppressors	53.5%	65.3%	66.7%	71.5%
Resisting cell death	75.9%	82.7%	86.9%	86.7%
Enabling replicative immortality	73.1%	90.9%	91.2%	91.5%
Inducing angiogenesis	73.9%	85.7%	74.8%	79.4%
Activating invasion and metastasis	72.5%	72.7%	82.0%	82.6%
Genomic instability and mutation	71.2%	69.2%	72.2%	81.7%
tumour promoting inflammation	69.9%	76.6%	81.6%	84.2%
Cellular energetics	78.1%	85.7%	76.6%	88.3%
Avoiding immune destruction	54.3%	71.8%	67.7%	75.8%
Average	69.2%	76.8%	76.6%	81.0%

Table 4.10: Comparison of test results using AUC

	SVM		CNN	
Hallmark	BoW	Rich	Base	Tuned
Sustaining proliferative signalling	88.6	88.9	92.1%	91.0%
Evading growth suppressors	87.9	91.7	94.8%	96.4%
Resisting cell death	92.4	95.5	97.1%	97.7%
Enabling replicative immortality	92.4	97.4	99.8%	99.5%
Inducing angiogenesis	94.7	98.4	97.9%	99.1%
Activating invasion and metastasis	96.0	94.0	97.8%	98.2%
Genomic instability and mutation	92.5	91.7	95.8%	97.0%
tumour promoting inflammation	92.7	95.9	98.3%	98.1%
Cellular energetics	99.1	99.6	99.5%	99.6%
Avoiding immune destruction	94.6	96.1	97.8%	99.1%
Average	93.1	94.9	97.1%	97.6%

4.2.5 Discussion

Our evaluation contrasts methods separated by two methodological divides: discrete, interpretable, hand-engineered features vs. continuous, opaque, automatically learned features for one, and convex optimization vs. gradient descent in a complex landscape with many local minima for the other. The choice between the SVM representing the former choices and the CNN representing the latter is not necessarily only a question of which performs better, but also of methodological fit, both to the broader machine learning framework and for the practitioners applying the approach.

A key point of interest in neural methods is feature learning, *i.e.* their capacity to learn complex models with minimal manual effort in feature engineering. As shown again in our experiments, a CNN taking only document text and word embeddings induced from unlabelled text as input can outperform an SVM with extensive

manually engineered features derived from sources such as syntactic analysis and named entity recognition. While the 3-4% point differences in AUC and F_1-score are positive results in favour of the CNN, the relative simplicity and generality of the model is arguably a greater advantage supporting the choice of the CNN over the feature-rich SVM—indeed, one might well argue that the most interesting of our results is that the basic general CNN without any task or domain adaptation only narrowly loses to the SVM in F_1-score, and outperforms it in terms of AUC. The CNNs can be more readily adapted to other tasks and carry much fewer technical requirements: while the handcrafted model utilises an NLP pipeline that uses separate tools for lemmatization, parsing, and named entity tagging in addition to the machine learning method, the CNN has no such external dependencies.

Potential shift to neural methods is not without its own issues. As detailed by Zhang and Wallace (2015), even the relatively simple CNN model presented here comes with a large number of hyperparameters and related modeling and optimization choices, many of which have task-specific optima, and the cost of training and evaluating large numbers of model variants can be prohibitive even on modern GPU-based systems. For machine learning researchers used to working with convex optimisation problems, the random elements involved in training neural network models can also be a source of frustration, and the need to account for variance from network initialisation and training also imposes additional computational costs.

4.3 Hierarchical text classification

Up to this point we have relied on OVR classification setup for multi-label classification, *i.e.* construct a binary classifier for each label in the taxonomy or ontology where all documents not belonging to the class are considered negative examples. This approach has two major drawbacks: first, it makes the hard assumption that the classes are independent which often does not reflect reality; second, it is more computationally expensive (albeit by a constant factor): if there are a very large number of classes, the approach becomes computationally unrealistic.

In this section, we describe a simple and computationally fast neural approach for multi-label classification with a focus on labels that share a structure, such as a hierarchy (taxonomy). This approach can work with established neural network architectures such as a CNN by simply initialising the final output layer to leverage the co-occurrences between the labels in the training data.

In multi-label text classification, input text can be associated with multiple labels (label co-occurrence). When the labels form a hierarchy, they share a hypernym–hyponym relation (Figure 4.5). When multiple labels are assigned to the text, if it is explicitly labelled by a subclass it must also implicitly include all of the its superclasses.

The co-occurrence between subclasses and superclasses as labels for the input text contains information we would like to leverage to improve multi-label classification using an ANN.

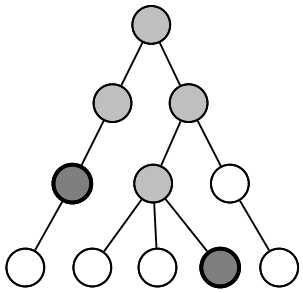


Figure 4.5: Hierarchical multi-label classification. Nodes represent possible labels that can be assigned to text: a dark grey node denotes an explicit label assignment and light grey denotes implicit assignment due to a hypernymy relationship with the explicitly assigned label.

4.3.1 Related work

There have been numerous works that focus on solving hierarchical text classification. Sun and Lim (2001) proposed top-down level-based SVM classification. More recently, Sokolov and Ben-Hur (2010); Sokolov *et al.* (2013) predict ontology terms by explicitly modelling the structure hierarchy using kernel methods for structured output space. Clark and Radivojac (2013) use a Bayesian network, structured according to the underlying ontology to model the prior probability.

Within the context of neural networks, Kurata *et al.* (2016) propose a scheme for initialising neural networks hidden output layers by taking into account multi-label co-occurrence. Their method treats some of the neurons in the final hidden layer as dedicated neurons for each pattern of label co-occurrence. These dedicated neurons are initialised to connect to the corresponding co-occurring labels with stronger weights than to others. They evaluated their approach on the *RCV1-v2* dataset (Lewis *et al.*, 2004) from the general domain, containing only flat labels. Their evaluation shows promising results. However, their applicability to the biomedical domain with more a complex set of labels that share a hierarchy remains an open question.

Chen *et al.* (2017) propose a CNN and recurrent neural network ensemble method that is capable of efficiently representing text features and modelling high-order label correlation (including co-occurrence). However, they show that their method is susceptible to overfitting with small datasets.

Cerri *et al.* (2014) propose a method for hierarchical multi-label text classification that incrementally trains a multi-layer perceptron for each level of the classification hierarchy. Predictions made by a neural network in a given level are used as inputs to the neural network responsible for the prediction in the next level. Their method was evaluated against several datasets with convincing results.

There are also several relevant works that propose the inclusion of multi-label co-occurrence into loss functions such as pairwise ranking loss (Zhang and Zhou, 2006) and more recent work by Nam *et al.* (2014), who report that binary cross-entropy

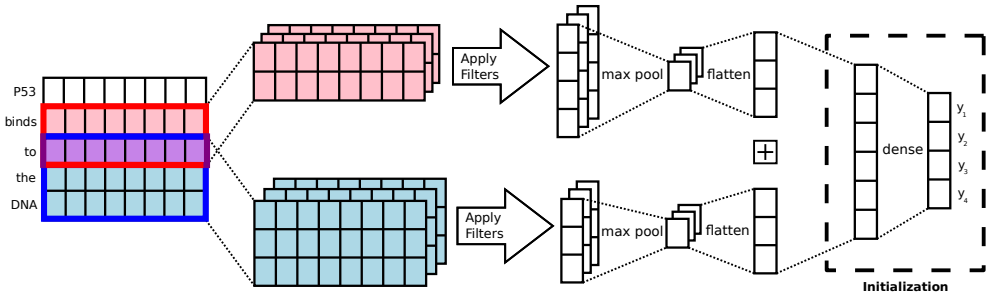


Figure 4.6: Initialisation of a CNN for hierarchical multi-label classification.

can outperform the pairwise ranking loss by leveraging ReLUs for nonlinearity.

4.3.2 Methodology

We utilise a similiar CNN architecture to that of Section 4.2, which consists of an initial embedding layer that maps input texts into matrices, followed by convolutions of different filter sizes and 1-max pooling, and finally a fully connected layer. The architecture is illustrated in Figure 4.6.The key difference is the final layer. In order to perform multi-label classification using this architecture, the final output layer uses logistic (sigmoid) activation function σ:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{4.6}$$

where x is the input signal. The output range of the function is between zero and one; if it is above a cut-off threshold T_σ (which is tuned by grid search on the development dataset) then the prediction y'_k for label y_k is positive. We use a binary cross-entropy loss function L:

$$L(\theta, (x, y)) = -\sum_{k=1}^{K} y_k \log(y'_k) + (1 - y_k) \log(1 - y'_k) \tag{4.7}$$

where θ is the model parameters and K is the number of classes.

As shown in Figure 4.6, the multi-label initialisation happens in output layer of the network. Figure 4.7 illustrates the initialisation process. The rows represent the units in the final hidden layer, while the columns represent the output classes.

The idea is to initialise the final hidden layer with rows that map to co-occurrence of labels in the training data. This can be implicit hypernymy relations between the labels, or explicit co-occurrence in the annotation. For each co-occurrence, the value ω is assigned to the associated classes and a value of zero is assigned to the rest.

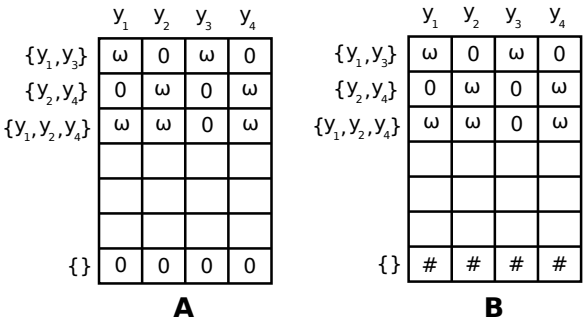


Figure 4.7: The two initialisation schemes: (A) initialising non label co-occurrence nodes with zero, (B) initialising non label co-occurrence with a random value (#) drawn from a uniform distribution.

Table 4.11: Our baseline model, based on Kim (2014) model hyperparameters.

Hyperparameter	Value
Word vector size	300
Filter sizes	3, 4, and 5
Number of filters	300 (100 of each size)
Dropout probability	0.5
Minibatch size	50
Input size (in tokens)	500 (documents), 100 (sentences)

The value ω is the upper bound of the normalised initialization proposed by Glorot and Bengio (2010), which is calculated as follows:

$$\omega = \frac{\sqrt{6}}{\sqrt{n_h + n_k}} \tag{4.8}$$

where n_h is the number of units in the final hidden layer and n_k is the number of units in the output layer (*i.e.* classes). This value was also successfully used by Kurata *et al.* (2016) in their initialisation procedure.

The motivation for this initialisation is to incline units in the hidden layer to be dedicated to representing co-occurrence of labels by triggering only the corresponding label nodes in the output layer when they are active.

The number of units in the final hidden layer can exceed the number of label co-occurrences in the training data. We must therefore decide what to do with the remaining hidden units. Kurata *et al.* (2016) assign random values to these units (shown in Figure 4.7 (B)). We will also use this scheme, but in addition we propose another variant: we assign the value zero for these neurons, so that the hidden layer will only be initialised with nodes that represent label co-occurrence.

We implement the neural network and the initialisation using Keras (Chollet, 2015). The hyperparameters for our model and baselines are those of Kim (2014), summarised in Table 4.11.

Table 4.12: Summary statistics for Tasks 1 and 2.

	Task 1: HoC taxonomy		Task 2: Exposure taxonomy	
	Document	Sentence	Document	Sentence
Train	1,303	12,279	2,555	25,307
Dev	183	1,775	384	3,770
Test	366	3,410	722	7,100
Total	1,852	17,464	3,661	36,177

Table 4.13: Jaccard similarity scores (expressed as percentages) between label pairs.

	Task 1: HoC taxonomy		Task 2: Exposure taxonomy	
	Document	Sentence	Document	Sentence
Avg	4.1	2.3	5.7	3.0
Max	49.3	49.4	45.7	42.5

We use `word2vec` embeddings trained on `PubMed` by Chiu *et al.* (2016).

4.3.3 Evaluation

We evaluate our on two hierarchical semantic classification tasks: Task 1: the Hallmarks of Cancer (HoC) taxonomy, Task 2: the Exposure taxonomy.

We split both datasets (by documents) into train, development (dev), and test splits in order to evaluate our methodology. Table 4.12 summarises key statistics for these splits.

We also measure the overlap in the data between pairs of labels. We use Jaccard similarity J to measure this overlap using the following equation:

$$J(A, B) = \frac{A \cap B}{A \cup B} \tag{4.9}$$

Where A and B are sets of instances labelled with these classes. Table 4.13 summarises the average and maximum pairwise Jaccard similarity between the labels in both tasks.

Table 4.13 shows that Evaluation Task 1 labels have slightly more overlap than those of Evaluation Task 2.

The large difference in values between document and sentence label overlap is due to the fact that documents have more labels per instance than sentences. The average score is much lower as most pair combinations would not have overlaps; where there is overlap it is typically significant (as shown by the Max row in Table 4.13).

We ascertain the performance of our approach under a controlled experimental setup. We compare two baseline models (described in the next section), and two variants of the initialization models corresponding to the two initialization schemes described in Figure 4.7. We will refer to the first scheme (allocating all units in

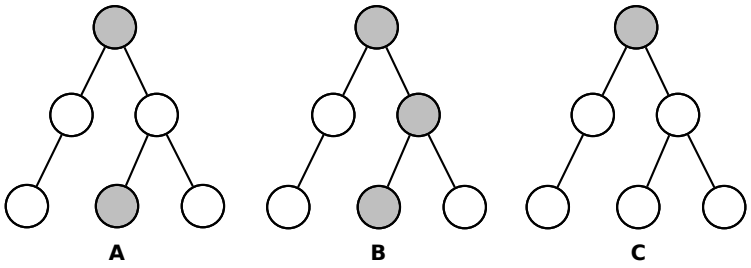


Figure 4.8: Illustrating post-processing label correction, with (A) showing the output prediction from the neural network model, (B) applying transitive correction, (C) applying retractive correction.

the final hidden layer to representing label co-occurrences and zeroing all other units) as INIT-A, and the second scheme (allocating a random value drawn from a uniform distribution for non co-occurrence hidden units) as INIT-B. We use the hyperparameters in Table 4.11 and data splits in Table 4.12 for all models.

We check the model's performance (F_1-score) on development data at the end of every epoch. We select the model from the best-performing epoch and train it until its performance does not improve for ten epochs.

We compare two baselines in our setup: OVR and multi-label baseline (MULTI-BASIC)

We train and evaluate K independent binary CNN classifiers (*i.e.* a single classifier per class with the instances of that class as positive samples and all other instances as negatives).

We train and evaluate a multi-label baseline based on Figure 4.6 without initialization of the final hidden layer. This enables us to directly compare the effect of the initialization step. As with the initialization models (INIT-A and INIT-B), we grid search the sigmoid cut-off parameter T_σ on the development data at the end of each epoch to be used with the selected best model on the test split.

The predicted output labels from all of our models can be inconsistent with respect to the label hierarchy: a subclass label might be positive while its superclass is negative, thereby contradicting the hypernmy relation (illustrated in Figure 4.8 (A)).

We can apply two kinds of post-processing corrections to the predicted labels in order for them to be well-formed. We call the first *transitive correction* (Figure 4.8 (B)), wherein we correct all superclass labels (transitively) to be positive. The alternative is *retractive correction* (Figure 4.8 (C)), where we ignore the positive classification of the subclass label, and accept only the chain of superclass labels (from the root), as long as they are well-formed.

We apply both of these post-processing correction policies to all of the models, and observe the effect on their performance.

Table 4.14: Performance results for Tasks 1 and 2. All figures are micro-averages expressed as percentages.

	Document			Sentence		
	P	R	F_1	P	R	F_1
Evaluation Task 1: Hallmarks of Cancer taxonomy						
OVR	77.8	51.7	62.1	56.8	30.7	39.9
MULTI-BASIC	71.0	71.6	71.3	42.0	71.9	53.0
INIT-A	73.4	76.9	75.1	42.7	70.6	53.2
INIT-B	68.3	83.4	75.1	40.1	72.2	51.6
Evaluation Task 2: Exposure taxonomy						
OVR	89.5	87.1	88.3	66.2	62.8	64.5
MULTI-BASIC	86.0	90.0	88.0	51.7	75.6	61.4
INIT-A	86.7	91.1	88.9	49.5	80.7	61.4
INIT-B	75.7	91.3	82.8	47.0	83.2	60.1

4.3.4 Results

We assess the performance of the models by measuring the standard precision (P), recall (R), and F_1-scores of the labels in the model using the OVR setup. Table 4.14 shows the micro-averaged scores across all labels for both tasks.

The results show that for Evaluation Task 1, all multi-labelled models significantly outperform the OVR model in F_1-score, which is explained by a very substantial improvement in recall. INIT-A outperforms all models in this task, particularly at the document level where there is a 5 point improvement over MULTI-BASIC.

The results for Evaluation Task 2 are more mixed. Overall, all models achieve a similar F_1-score at the document level. However, there is a clear improvement in recall at the cost of lower precision when compared to OVR. The best performing model at the document level is INIT-A. On the sentence level, OVR seems to outperform all multi-label models by a good margin. This indicates that the multi-label approach did not aid sentence-level classification in this particular task.

The figures in Table 4.14 do not show a complete picture as the interactions between the labels are not taken into account.

We can observe the proportion of the number of labels assigned to each instance by the classifiers, and compare these proportions to the annotated gold standard test data. Figure 4.9 shows this distribution for each classifier. We can see in Figure 4.9 that the overall distributions for all sentence-level classifiers (for both tasks) are closer to the gold standard distribution (compared to document-level). This is due to the fact that most sentences have no assigned labels. For Evaluation Task 2, the classifiers tend to assign more labels than the gold standard.

Document-level classification shows two outliers. For Evaluation Task 1, we observe that OVR disproportionately assigns exactly one label per document compared to gold standard (where documents have two to three labels on average). In

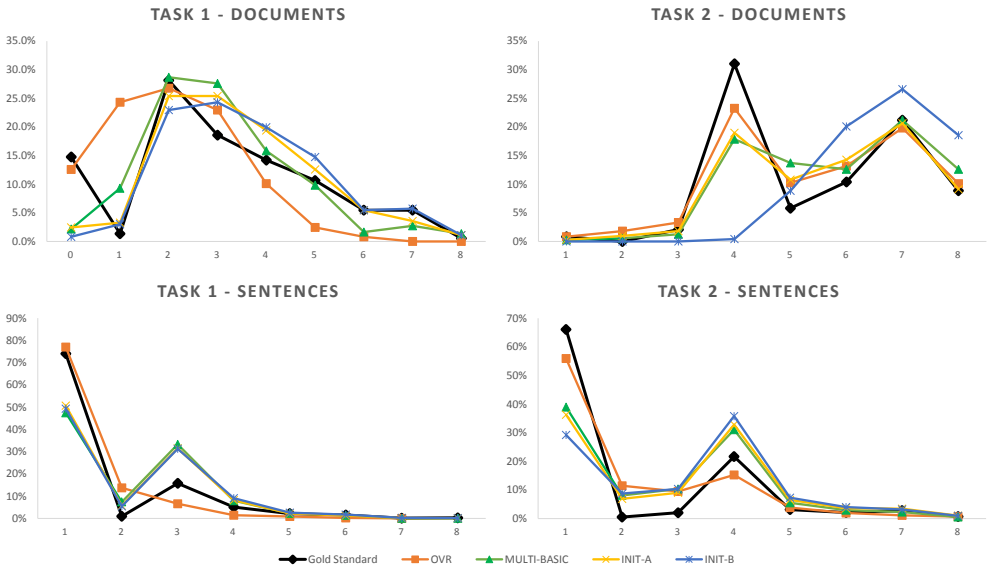


Figure 4.9: The distribution of instances according to the number labels per instance. The number of labels per instance (x-axis), and y-axis is the proportion of instances in the test dataset that have that number of labels. The black line indicates the distribution of the gold standard annotation (*i.e.* ground truth).

Table 4.15: The proportion (%) of exact matches.

	Task 1: HoC taxonomy		Task 2: Exposure taxonomy	
	Document	Sentence	Document	Sentence
OVR	18.0	67.9	43.4	61.7
MULTI-BASIC	26.2	59.3	40.9	54.2
INIT-A	33.9	65.9	45.6	53.1
INIT-B	31.3	62.6	12.7	49.7

Evaluation Task 2, INIT-B assigns more labels per document than the gold standard (and every other model).

In addition to looking at the number of labels per class, we also measure the proportion of exact label matches that each model predicts as shown in Table 4.15.

For document classification in Evaluation Task 1, INIT-A outperforms all models, while OVR significantly underperforms. However, OVR performs significantly better than all other models when classifying sentences when considering exact matches only.

Finally, we look at how consistent (well-formed) the predictions output by each model are. We do this by running the post-processing label correction policies illustrated in Figure 4.8. Table 4.16 summarises these results.

For Evaluation Task 1, OVR shows the largest variance after the application of any method of correction, whereas no multi-labelled model shows this variation. This indicates that the post-processing corrections had little effect on the predicted

Table 4.16: Post-processing label correction. O is the predicted output, T is transitive correction, and R is retractive correction. All figures are micro-averaged F_1-scores expressed as percentages.

	Document			Sentence		
	O	T	R	O	T	R
Evaluation Task 1: Hallmarks of Cancer taxonomy						
OVR	62.1	63.9	60.6	39.9	42.2	37.5
MULTI-BASIC	71.3	71.3	71.2	53.0	53.0	53.0
INIT-A	75.1	75.0	75.2	53.2	53.2	53.3
INIT-B	75.1	74.9	75.3	51.6	51.5	51.6
Evaluation Task 2: Exposure taxonomy						
OVR	88.3	88.4	88.2	64.5	65.3	63.3
MULTI-BASIC	88.0	87.7	88.1	61.4	61.3	61.7
INIT-A	88.9	88.7	89.0	61.4	61.3	61.5
INIT-B	82.8	82.8	82.8	60.1	59.8	60.4

results as they were already well-formed. For Evaluation Task 2, there is very little variance for all multi-labelled models, with only a slight change for OVR.

4.3.5 Discussion

The strength of using the hidden-layer initialisation for multi-label classification lies in leveraging the co-occurrence between labels. Naturally, if such co-occurrences are relatively rare in the dataset, then this approach becomes less effective. This implies that this approach is especially attractive for hierarchical multi-label classification, because of the implicit hypernym–hyponym relations between the labels, which by definition guarantees co-occurrence of labels in the datasets. The superclass labels must be included when labelling a given example in order to model the hierarchical nature of the labels.

Another key strength of this approach is its low computational cost, which is only proportional to the size of the input text, and the number of label co-occurrences.

However, when there is a large amount of training data, the number of label co-occurrences can be larger than the number of the hidden units. In such a case, one possible option is to select an appropriate subset of label co-occurrences using a certain criteria such as the frequency in the training data. For the datasets used in this paper, this was not necessary.

Overall, the results of the evaluation show that initialising the model using only label co-occurrences (INIT-A) generally produced a higher performance compared to the other models, including the random initialization of remaining hidden units in the final hidden layer (the INIT-B model) as proposed by Kurata *et al.* (2016). However, there was one key exception in Evaluation Task 2—sentence-level classification,

where the OVR model achieved the best results.

Both variants of the initialisation models investigated here achieved generally positive results when the scope of text is larger (*i.e.* documents), where there are more labels assigned per text instance. However, due to time and computational constraints, this initialisation method was not fully utilised as we could only investigate its performance under a closed set of hyperparamaters for the CNN model. It may be possible for this approach to yield even better results if further parameters are included in the CNN models (*e.g.* more filters and filter sizes). It is also important to note that collectively the OVR models have much more parameters than any of the multi-label models in our experiment setup, and therefore they have a higher capacity to capture correlations. In spite of this, the multi-label models have largely outperformed the OVR model.

There are many tasks in the biomedical domain that require the assignment of one or more labels to input text. These labels often exists within some hierarchical structure (such as a taxonomy). We believe that this approach shows promising potential for benefiting these tasks.

4.4 Chapter Summary

In this chapter, we investigated the use of neural methods to the task of semantic text classification. Neural methods have the potential to greatly reduce the need for feature engineering, and allow for more portability and transfer between subdomains.

We first investigated a method that jointly learns a multi-level embedded representation for the target class labels, as well as words, sentences, and documents in the same vector space. The intuition behind multi-level embeddings is that each level captures slightly different topical semantics. We employ these embeddings to produce three types of features that are relatively efficient to extract and much easier to port than handcrafted features. We have shown how these feature can attain good performance with standard classification algorithms such as SVMs and with semi-supervised classification where the decision boundaries are not learned from labelled data.

Next, we apply the use of neural models for automatic feature learning. We demonstrated that a CNN model taking only the input text and word representations induced from unannotated general-domain text as input can achieve competitive performance on the Hallmarks of Cancer data compared to feature rich SVM baseline with rich manually engineered features.

We have so far used the CNN and SVM classifiers for multilabel classification using a one-*vs*.-rest setup that assumes a flat independent label structure. The main drawbacks to this approach are that dependencies between classes are not leveraged in the training and classification process, and the additional computational cost of training a classifier for each class. We then implemented a method for multilabel

classification that initialises a neural network model final hidden layer to leverage label co-occurrence.

We evaluated this approach using two hierarchical multilabel classification tasks (HoC and Exposure taxonomies) using both sentence and document level classification. The experimental results for both tasks show that overall, this method achieved better results than all of the other models, with the exception of sentence-level classification for the Exposure taxonomy.

Chapter 5

Real-world applications

In the previous two chapters, we described the task of semantic text classification and detailed several methods and models that were trained and evaluated chiefly on two cancer-related classification tasks. This chapter will focus on applying these classification methods to TM tools for cancer researchers. We present two such tools. The first is a tool that extracts correlations between the HoC and an input text query on PubMed abstracts. The second tool is a literature-based discovery system for cancer. We describe these tools, their implementation, and their evaluation.

5.1 Cancer hallmark analytics tool

In Chapter 3, we described the Hallmarks of Cancer (HoC)—a set of ten principles that concisely describe the inception and proliferation of cancer—and a taxonomy developed by cancer researchers at Karolinska Institutet. In Section 3.2.2, we introduced a methodology for classifying PubMed abstracts according to the HoC taxonomy. In evaluating the methodology, we described several case studies that demonstrated its validity compared to established biomedical knowledge, and also its usefulness for experts.

The processs of literature review is a vital part of biomedical research (Payne *et al.*, 2007). It would be useful for cancer researchers to retrieve and organise text according the HoC on a large scale. Currently, cancer researchers use IR systems such as PubMed for keyword-based queries. However, due to the scope and complexity of cancer literature the number of keywords, their synonyms, and possible combinations exceeds what researchers can memorise and manage. Overly complex queries can fail to achieve a satisfactory level of precision and recall. As described in Section 3.2.1, each hallmark can be associated with several keywords and phrases which, when found in literature, represent good indicators for the presence of the hallmarks in text.

Here we describe a cancer hallmark analytics tool (CHAT), which was designed to support the process of literature review in the field of cancer research. The

propose of CHAT is to text mine PubMed by finding association for a given text query and cancer hallmarks.

CHAT can be accessed *via* the following URL:

http://chat.lionproject.net/

5.1.1 Functionality overview

CHAT automatically organises and classifies the literature with good accuracy, and identifies key correlations which are in line with the existing knowledge. Developed in close collaboration with cancer researchers, CHAT can be of great use for classifying scientific literature by cancer hallmarks and associated biological processes.

We integrated PubMed documents and the hallmark sentence classification generated by our NLP pipeline into a database, and created a web-based interface (shown in Figures 5.1 and 5.2) in close consultation with cancer researchers. The interface allows users to analyse the distribution of a search query of interest with respect to the hallmarks using multiple visualisations. Several options are provided for user metrics: raw counts, conditional probability values (*i.e.* the probability of the sentence being assigned the hallmarks given the query), point-wise mutual information (PMI), and normalised point-wise mutual information (NPMI), which are calculated as follows:

$$P(p \mid q) = \frac{P(h,q)}{P(q)} \tag{5.1}$$

$$\text{PMI} = \log\left(\frac{P(h,q)}{P(h)\,P(q)}\right) \tag{5.2}$$

$$\text{NPMI} = \frac{\text{PMI}\,(h,q)}{-\log\left(P(h,q)\right)} \tag{5.3}$$

where h and q denote a given hallmark and a search query. The user interface (UI) enables the user to explore the source data and to assess the evidence for specific associations between query terms and the hallmarks (Figure 5.3). In addition, the UI allows the user to compare two queries on the same graph (mirrored bar graph) as well as to examine the statistical significance results of the comparison (illustrated in Figure 5.3). In the comparison screen, the tool automatically employs as a statistical test either the Fisher-exact test or χ^2 test followed by a Bonferroni correction. We use the Fisher-exact test if the expected frequency is less than five as typically recommended by statisticians (McDonald, 2009). CHAT also allows the user to download the data displayed in the graph for further analysis.

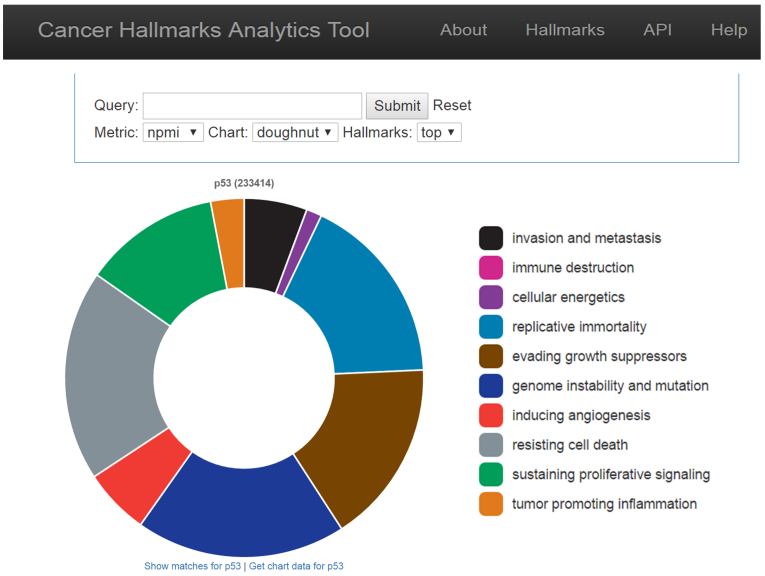


Figure 5.1: CHAT visualises the hallmarks distribution for an input query (in this example the query is "p53"). There are several visualisation options; in this example, the hallmarks are depicted in a ring akin to the original Hallmarks of Cancer publication (Hanahan and Weinberg, 2000).

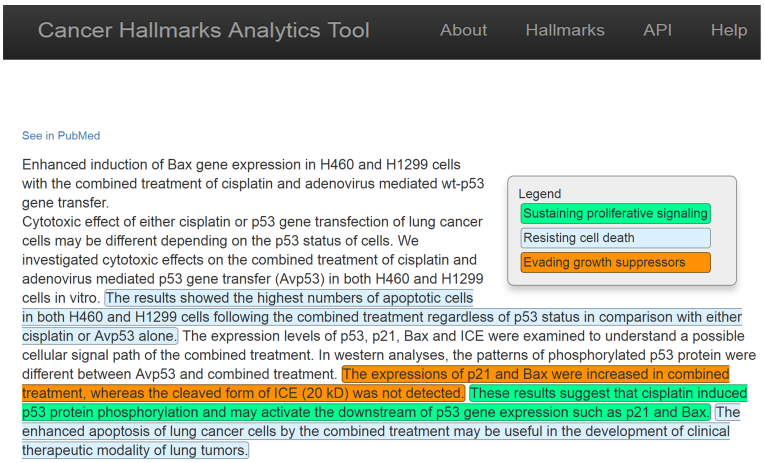


Figure 5.2: CHAT allows the user to explore individual abstracts (*i.e.* to drill down for evidence sentences classified by CHAT) and visualises the hallmark labels appearing in the text.

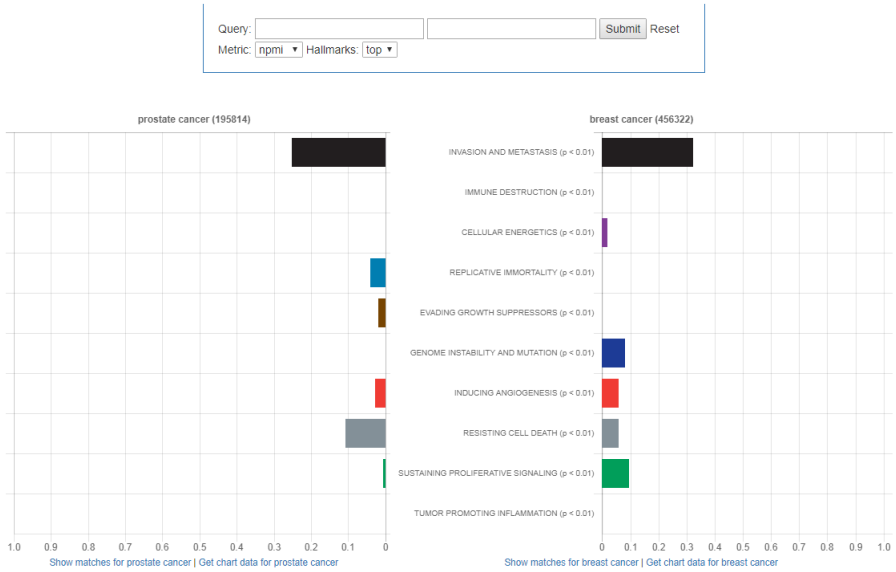


Figure 5.3: CHAT allows the user to compare the hallmark distribution profile of two queries (in this example, the queries are "prostate cancer" and "breast cancer", and calculates the statistical significance between the distributions.

5.1.2 Implementation

In implementing CHAT, we first used `Lucene`[33]—a state-of-the-art indexing and text search engine—to index all of `PubMed` (2016 release), and also the hallmark label predictions generated by our NLP pipeline. We then created a web interface using the Python `Flask`[34] framework to allow for flexible querying of the data, and implemented client-side visualisation of results using the `Chart.js`[35] JavaScript charting library.

Our methodology is based on the supervised NLP pipeline introduced in Section 3.2.2, and is shown in Figure 5.4. However, due to computational requirements demanded by scaling our methodology to process the entire `PubMed` abstract corpus,[36] we need to ensure that the computational time needed for our pipeline is not prohibitive. As a result, we removed from the pipeline the GR feature type, which requires significant computational resources. We also introduced new features to the pipeline, and streamlined the tools in the pipeline to allow faster execution time.

The CHAT pipeline extracts seven types of semantic and syntactic features from scientific literature: Lemmatised Bag of Words (LBOW), *n*-grams (bigrams and trigrams), verb classes (VCs), named entities (NEs), Medical Subject Heading (MeSH), chemicals list (CHEM), semantic distance (SD).

[33] http://lucene.apache.org/
[34] http://flask.pocoo.org/
[35] http://chartjs.org/
[36] `PubMed` 2016 baseline.

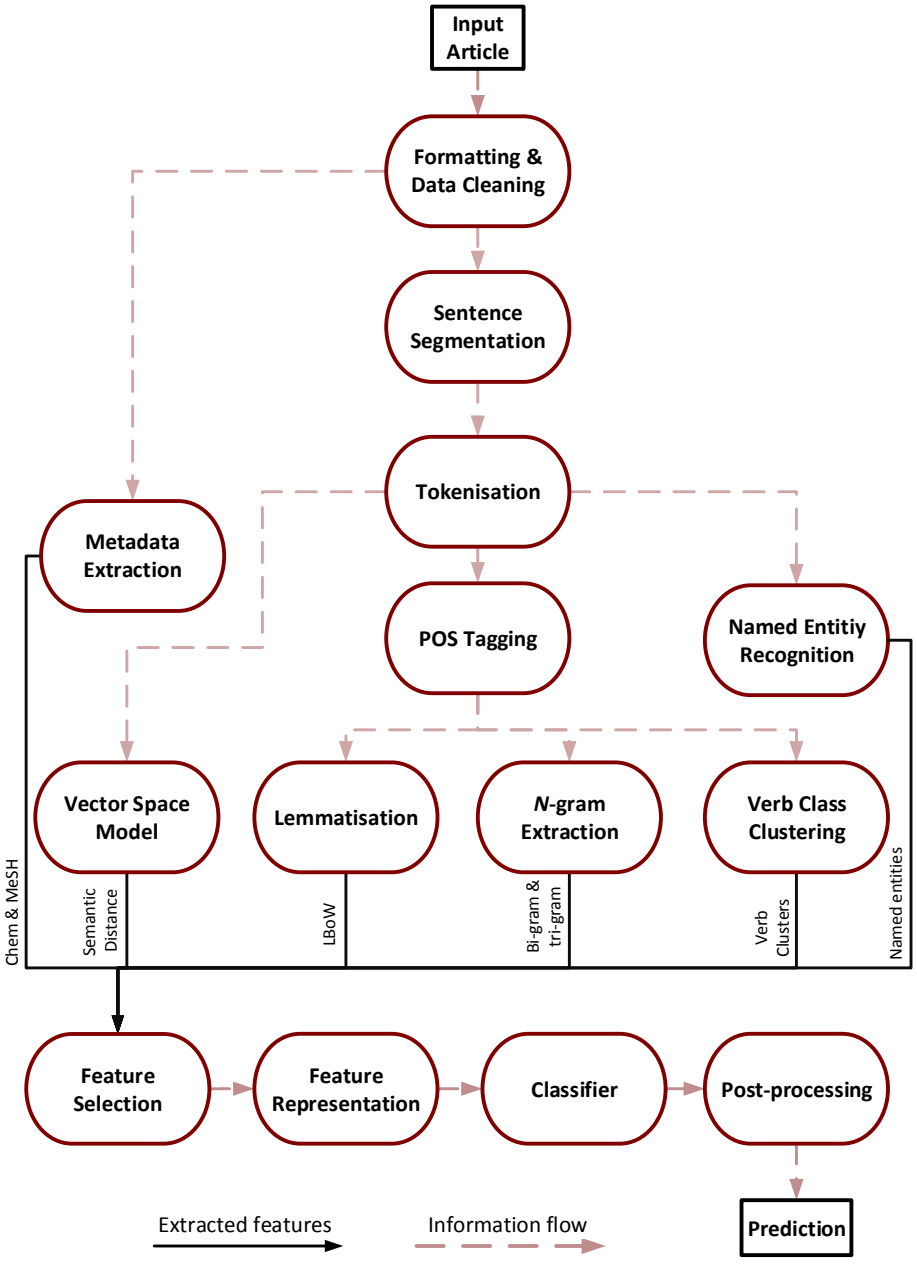


Figure 5.4: CHAT NLP pipeline

With the exception of SD and *n*-grams, all of the features were already utilised in Section 3.2.2. However, we still need to reconsider our features for sentences rather than abstracts. The most notable change is for metadata features (MeSH and CHEM) as both of these feature types apply only to documents and not sentences; however, they still provide information that is beneficial for sentence classification. Here, we assign all sentences in a given abstract with the abstract's metadata features.

n-gram features are introduced in this pipeline as they are inexpensive to compute and can be beneficial for text classification. We use bigram and trigram features only.

The SD feature type is essentially identical to the class-sim feature used in Section 4.1.2—*i.e.*, We construct a semanticvector space model (VSM), using the multilevel embedding representation described in Section 4.1.2. Here, we use only class-, sentence-, and word-levels for our representation. We then use cosine similarity to measure the distance between words occurring in the sentence and a given hallmark label.

We use the `GENIA tagger`[37] (Kulick *et al.*, 2004; Tsuruoka *et al.*, 2005; Tsuruoka and Tsujii, 2005) to perform the POS tagging, lemmatisation, and NER steps of the pipeline.

We apply feature selection: features that are deemed too rare or too common in the annotated corpus are filtered out, so that only the most discriminating ones are used. This typically means a minimum threshold value of five occurrences, while the maximum threshold varies greatly depending on the feature type (usually a value greater than 500 occurrences). This improves accuracy and reduces training time. This procedure is done separately for each of the hallmarks, *i.e.*, we only select the features in the corpus that occur in abstracts annotated with the given hallmark. Therefore, each classifier has a unique set of selected features. The number of features for each hallmark after feature selection is given in Table A.4.

The features are represented in a sparse binary format for each sentence, with a value of '1' indicating that the given sentence contains this feature. The binary features are then input into 37 SVM classifiers with linear kernels that label each sentence with a binary label indicating its relevance to one of the 37 labels in the hallmark taxonomy. Each of the classifiers is trained and executed independently to allow for mutually nonexclusive multilabel classification (*i.e.* we used an OVR training scheme).

We enforce the HoC hypernymy relationships in our classification—*i.e.*, we consider subclass labels as positive examples when we are classifying their parent nodes. For example, when classifying the hallmark "resisting cell death", the sentences annotated with the subclass "apoptosis" would be considered positive examples for "resisting cell death". We achieved this by applying our retractive postprocessing correction from Section 4.3.3.

We use `Scikit-learn`[38] (Pedregosa *et al.*, 2011) to implement the SVM classifier step of the pipeline. Since we have heavily imbalanced classes (far more negative

[37] http://www.nactem.ac.uk/GENIA/tagger/
[38] http://scikit-learn.org/

examples than positive ones), we apply inverse proportional class weighting to adjust for this imbalance.

5.1.3 Evaluation

We first describe the intrinsic evaluation of the NLP pipeline using standard methods and metrics. We then assess the functionality and the practical usefulness of CHAT with several case studies on cancer research.

For the intrinsic evaluation, we use a nested cross-validation setup. The data is divided into four folds—*i.e.*, the model is trained with 75% of the data and tested with the remaining 25%, and this split configuration is rotated four times for full coverage of the dataset. The size of folds was selected based on the sparsity of the test data. Within the 75% of the training data, we also perform another step of cross-validation for parameter tuning of the SVM kernels. Here we apply five-fold cross-validation, in which we train with 80% of the data (for a given parameter configuration) and test on the remaining 20%.

We observe in Table 5.2 that the classifiers, on average, exhibit good accuracy and F_1-score. The macro-average F_1-score for the ten hallmarks is 54.9%, the micro-average is 54%, and accuracy is 96.3%; the average F_1-score for the entire taxonomy is approximately 52% with an accuracy of 97.9%. The classifiers perform well when considering the inter-annotator agreement ($\kappa = 0.67$ for the ten hallmarks, and $\kappa = 0.61$ for the entire taxonomy), as well as the fact that on average, only about 10% of the sentences in the corpus are labelled with a hallmark.

The performance is lower for some of the leaf subclasses of the taxonomy (*e.g.*, immune response). This is because of the low number of positive examples associated with these subclasses in the annotated corpus (Table 5.1), causing the set of discriminating features extracted by our pipeline to be sparse.

We also evaluated the usefulness of our features using leave-one-out feature analysis; the results are presented in Table 5.3. Overall, the results of the analysis is consistent with the earlier HoC experiments in Section 4.1.3. The results show that the most influential feature type was LBOW, followed closely by SD—both lead to a significant decrease in performance when removed. VC was the weakest feature type. On average, it resulted in a marginal performance improvement when removed; however, it was still a useful feature for many hallmark classes.

Table 5.1: Intrinsic evaluation results for the CHAT classifiers.

Hallmark	# Annotated	# Classified	# Features
Activating invasion and metastasis	667	943 054	5218
Invasion	282	271 211	3202
Metastasis	317	591 214	3383
Avoiding immune destruction	226	651 044	2237
Immune response	152	465 785	1696
Immunosuppression	70	70 881	1035
Cellular energetics	213	84 204	2006
Glycolysis / Warburg effect	195	48 772	1870
Enabling replicative immortality	295	49 223	2323
Immortalization	111	6407	1193
Senescence	185	39 298	1620
Evading growth suppressors	366	579 810	4237
Deregulating cell cycle checkpoints	251	144 562	2908
Cell cycle	238	139 071	2747
Evading contact inhibition	118	273 566	1864
Genomic instability and mutation	768	1 397 318	5675
DNA damage	371	193 566	3522
Adducts	97	37 599	918
Strand breaks	121	30 174	1515
DNA repair mechanisms	213	95 510	2483
Mutation	215	826 072	2042
Inducing angiogenesis	358	308 574	2854
Deregulating angiogenesis	350	287 854	2776
Angiogenic factors	171	118 377	1696
Resisting cell death	832	863 918	7141
Apoptosis	610	594 979	5841
Autophagy	157	33 845	1098
Necrosis	108	198 429	1682
Sustaining proliferative signalling	993	811 719	7479
Cell cycle	320	141 941	3631
Growth factors growth promoting signals	323	224 980	3407
Downstream signalling	138	69 880	1952
Receptors	345	278 561	3558
Tumor promoting inflammation	518	1 145 524	4659
Immune response	78	117 320	1017
Inflammation	452	928 736	4445
Oxidative stress	241	220 979	2605

Table 5.2: CHAT NLP pipeline intrinsic evaluation results.

Hallmark	P (%)	R (%)	F_1 (%)	A (%)
Activating invasion and metastasis	54.5	75.9	63.4	96.7
Invasion	50.1	62.4	55.6	98.4
Metastasis	53.8	71.3	61.3	98.4
Avoiding immune destruction	32.2	59.3	41.7	97.9
Immune response	23.2	38.2	28.9	98.4
Immunosuppression	51.5	50.0	50.7	99.6
Cellular energetics	45.8	79.8	58.2	98.6
Glycolysis / Warburg effect	47.1	74.9	57.8	98.8
Enabling replicative immortality	59.0	85.8	69.9	98.8
Immortalization	61.7	73.9	67.2	99.5
Senescence	62.8	85.9	72.6	99.3
Evading growth suppressors	39.0	62.0	47.9	97.2
Deregulating cell cycle checkpoints	32.9	49.4	39.5	97.8
Cell cycle	33.6	46.6	39.1	98.0
Evading contact inhibition	68.5	83.1	75.1	99.6
Genomic instability and mutation	36.3	72.7	48.4	93.2
DNA damage	39.2	70.9	50.5	97.0
Adducts	59.2	62.9	61.0	99.6
Strand breaks	32.9	47.1	38.8	99.0
DNA repair mechanisms	39.2	61.0	47.7	98.4
Mutation	36.8	61.4	46.0	98.2
Inducing angiogenesis	40.2	66.2	50.0	97.3
Deregulating angiogenesis	40.3	65.4	49.9	97.4
Angiogenic factors	42.5	53.2	47.3	98.8
Resisting cell death	56.5	82.1	66.9	96.1
Apoptosis	60.7	79.8	69.0	97.5
Autophagy	61.4	79.0	69.1	99.4
Necrosis	66.9	76.9	71.6	99.6
Sustaining proliferative signalling	36.5	67.1	47.3	91.5
Cell cycle	48.5	60.3	53.8	98.1
Growth factors growth promoting signals	27.0	35.3	30.6	97.0
Downstream signalling	41.2	29.0	34.0	99.1
Receptors	33.3	54.5	41.4	96.9
Tumor promoting inflammation	40.1	66.6	50.1	96.1
Immune response	25.0	34.6	29.0	99.2
Inflammation	42.4	66.8	51.8	96.8
Oxidative stress	46.1	61.4	52.7	98.5
Macro-average	45.1	63.6	52.3	97.9
Micro-average	43.7	66.8	52.8	97.9

Table 5.3: CHAT leave-one-out feature analysis results. All figures are F_1-scores expressed as percentages.

Hallmark	All	LBOW	n-gram	VC	NE	MeSH	Chem	SD
Activating invasion and metastasis	63.4	39.9	64.2	62.9	63.7	62.0	63.8	51.4
Invasion	55.6	34.4	51.7	56.4	55.6	58.4	58.2	47.8
Metastasis	61.3	29.6	61.4	61.9	61.3	63.0	60.3	53.4
Avoiding immune destruction	41.7	23.6	42.1	44.0	43.8	41.8	44.7	33.4
Immune response	28.9	20.3	28.3	33.5	30.7	27.1	33.8	25.1
Immunosuppression	50.7	42.5	48.0	55.4	52.5	48.3	52.2	43.6
Cellular energetics	58.2	41.0	55.4	57.2	57.7	60.9	57.7	51.8
Glycolysis/Warburg effect	57.8	40.4	53.8	57.8	57.0	51.9	58.1	49.7
Enabling replicative immortality	69.9	42.1	60.1	69.2	69.0	69.4	68.2	57.3
Immortalization	67.2	30.0	65.0	67.7	65.3	67.2	65.0	55.1
Senescence	72.6	63.9	68.2	71.9	72.7	66.1	72.2	65.3
Evading growth suppressors	47.9	39.3	45.1	49.7	48.5	45.4	50.2	39.7
Deregulating cell cycle checkpoints	39.5	30.9	34.7	39.6	39.6	40.1	42.4	30.8
Cell cycle	39.1	30.6	36.7	40.0	39.5	38.8	40.7	33.6
Evading contact inhibition	75.1	72.4	65.2	76.9	74.3	75.6	75.2	72.6
Genomic instability and mutation	48.4	36.8	46.8	48.4	48.0	50.2	48.4	38.2
DNA damage	50.5	36.7	48.2	50.0	49.3	50.1	50.3	39.9
Adducts	61.0	29.7	58.5	62.9	57.4	60.8	57.1	51.2
Strand breaks	38.8	20.9	35.5	35.6	36.4	38.5	40.0	34.9
DNA repair mechanisms	47.7	31.3	41.2	47.3	47.1	46.9	49.0	41.0
Mutation	46.0	24.8	44.0	46.8	46.0	44.5	45.5	40.9
Inducing angiogenesis	50.0	38.3	51.7	49.3	50.3	50.3	50.4	40.0
Deregulating angiogenesis	49.9	38.9	50.2	49.4	50.2	50.5	50.2	39.9
Angiogenic factors	47.3	36.8	48.0	44.6	46.8	45.9	48.2	36.4
Resisting cell death	66.9	44.8	66.3	66.9	66.3	69.5	67.1	50.2
Apoptosis	69.0	48.9	63.7	70.3	69.3	60.5	69.8	53.1
Autophagy	69.1	49.2	66.9	68.8	69.8	70.5	68.7	52.5
Necrosis	71.6	50.9	68.9	74.2	70.7	64.7	71.7	52.0
Sustaining proliferative signalling	47.3	40.6	42.4	47.6	46.7	45.7	47.8	36.4
Cell cycle	53.8	39.2	50.2	53.4	54.5	49.9	54.0	40.9
Growth factors growth promoting signals	30.6	26.0	28.7	31.1	29.2	30.8	31.1	23.5
Downstream signalling	34.0	23.8	29.4	33.9	32.0	35.7	33.3	26.6
Receptors	41.4	29.7	40.6	42.3	41.1	40.5	41.4	33.9
Tumor promoting inflammation	50.1	34.0	50.1	50.6	49.9	51.3	51.2	38.6
Immune response	29.0	10.7	24.1	30.1	27.9	22.2	27.3	24.1
Inflammation	51.8	35.1	51.6	52.8	52.2	47.1	52.8	43.0
Oxidative stress	52.7	40.2	53.7	54.8	52.0	54.9	54.2	43.7
Average	52.3	36.4	49.7	52.8	52.0	51.3	52.8	43.0

Case studies

To evaluate the practical usefulness of CHAT for cancer research, we present here four example case studies. Our aim is to test whether CHAT can classify the broad and varied range of text accurately into the relevant classes of the HoC taxonomy. The results for each case study are described below and are illustrated in Figure 5.5.

Case study 1: Lung cancer and Cisplatin We used CHAT to analyse `PubMed` literature on lung cancer and the commonly used drug to treat this cancer, Cisplatin (Figure 5.5-A). Cell invasion and metastasis is the most common hallmark associated with lung cancer in the classified literature, which is in line with existing knowledge (Nguyen *et al.*, 2009). Cisplatin interferes with DNA replication, which kills cells through apoptosis (Wang and Lippard, 2005). Our literature analysis shows that apoptosis is the most frequent hallmark associated with Cisplatin, which demonstrates the ability of the tool to efficiently and accurately classify the literature. Furthermore, Cisplatin studies have a hallmark profile more similar to that of lung cancer than that of colorectal cancer. This might reflect the more common use of Cisplatin in lung cancer treatment.

Case study 2: Aspirin and colorectal cancer Low-dose aspirin treatment is used to prevent colorectal cancer. As for lung cancer, the automatic literature analysis on colorectal cancer shows that invasion and metastasis is the most common cancer hallmark in the classified literature (Figure 5.5-B). The literature profile of aspirin shows inflammation as the most common cancer hallmark associated with aspirin, which is in line with the fact that targeting inflammation is one of the key mechanisms by which aspirin acts to prevent colorectal cancer (Drew *et al.*, 2016).

Case study 3: Growth factor EGF and VEGF

Epidermal growth factor (EGF) and vascular endothelial growth factor (VEGF) are important in human cancers. EGF stimulates cell proliferation by binding to its receptor EGFR (Normanno *et al.*, 2006), whereas VEGF and its cognate receptor play a central role in angiogenesis (Zhao and Adjei, 2015). The CHAT classification shows that sustaining proliferative signalling and angiogenesis are the most common hallmarks associated with EGF and VEGF, respectively, in literature (Figure 5.5-C).

Case study 4: Housekeeping genes TBP and GAPDH

Housekeeping genes (HKGs) are often used as reference genes when studying alterations in gene expression as a response for instance, to cellular stresses (Iyer *et al.*, 2017). HKGs are expected to maintain constant expression levels in different conditions. Here we have analysed two HKGs:TATA-Box binding protein (TBP)[39] and glyceraldehyde 3-phosphatase dehydrogenase (GAPDH). The CHAT classification shows that the classical GAPDH significantly associated with cellular energetics and Warburg effect, while TBP does not show any significant association with any of the hallmarks (Figure 5.5-D). This data is in line with the experimental findings showing that HKGs may be affected and respond differently depending on stress conditions (Iyer *et al.*, 2017).

[39] Also known as Goldberg-Hogness box.

5.1.4 Discussion

Comprehensive and efficient use of existing scientific knowledge is critically important for generating novel ideas for cancer research. Scientists working in this area use systems such as `PubMed` to gather existing information of relevance to their research. However, given the wide range and complexity of cancer-related scientific data and the number of relevant keywords, their synonyms and potential combinations exceeds what a scientist can reasonably memorise and handle.

The motivation behind CHAT is to identify and semantically organise cancer-related scientific literature in meaningful categories. This often needed by cancer researchers for thorough review of literature and identification of the molecular processes involved in cancer development. CHAT analyses and classifies cancer-related literature based on the HoC taxonomy, and was developed in collaboration with end-users (cancer researchers). CHAT enables the users to immediately analyse the correlation between any query term and the hallmarks and the associated process according to the HoC taxonomy. Furthermore, the tool provides a variety of statistical analyses and visualisations of the hallmark annotations in their original sentence context.

CHAT performs much finer-grained classification at the level of sentence than the methodology presented in Chapter 3 (Section 3.2.2), as this enables us to capture co-occurrences between the search query and the classified hallmark at a more granular text window, thereby extracting less noisy correlations. However, in comparison with abstract-level classification, sentence-level classification is a more difficult NLP problem. The much smaller context window available as input to the classifier tends to reduce the classifier's accuracy. However, this reduction in accuracy is a good tradeoff compared to the gains we achieve by using a large classification window. This is evidently true as less than 10% of the sentences are associated with any hallmarks in the annotated data, *i.e.*, most sentences will not contain any hallmark-related information, and therefore standard co-occurrence measurements such as PMI would be too noisy if used with abstract-level classification.

The NLP pipeline performed with promising accuracy, particularly given the challenges of sentence-level classification. The conducted case studies focused on cancer types, therapeutics, growth promoting proteins and housekeeping genes, showing that CHAT identifies correlations that agree with established scientific consensus. The tool proved useful for classifying cancer-related text and text mining associated biological processes, with a simple search query on cancer types, intrinsic or extrinsic factors, genes, and therapeutics.

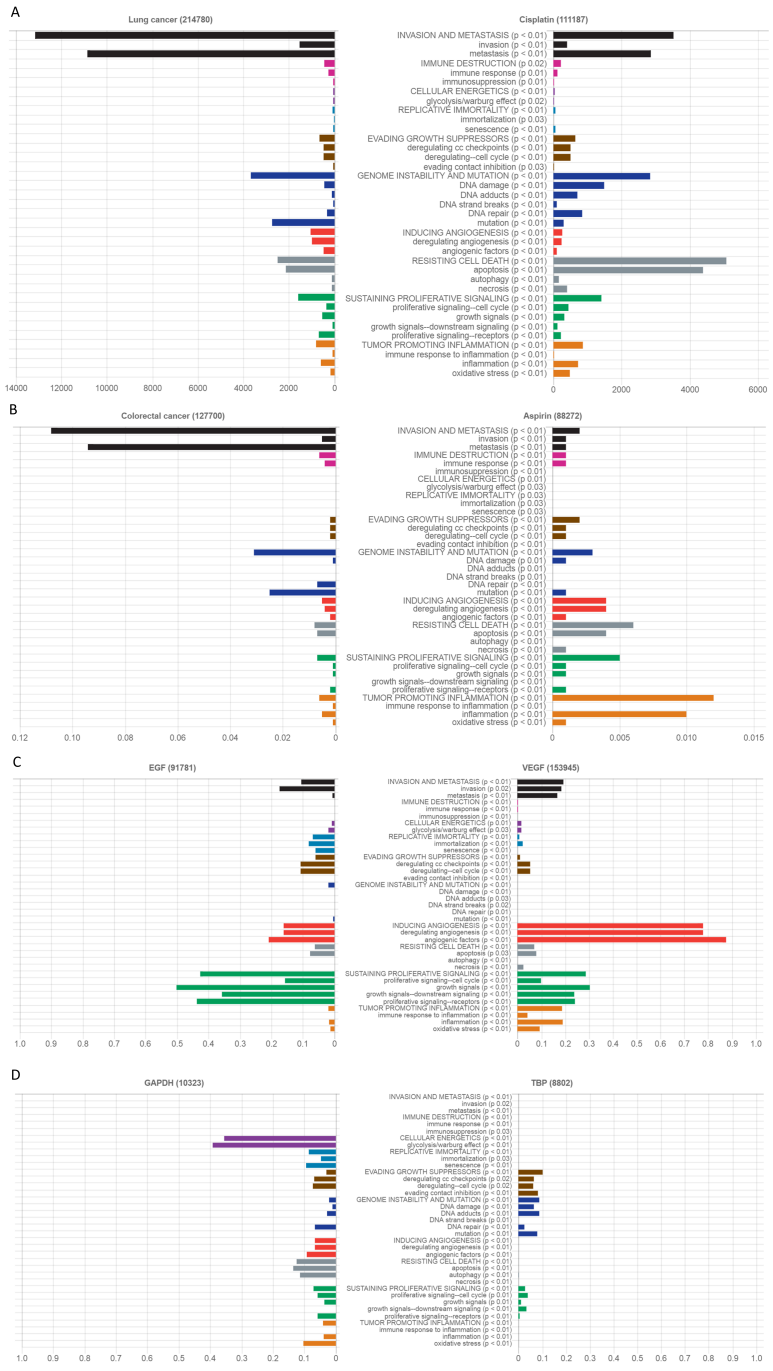


Figure 5.5: CHAT classification of `PubMed` literature according to HoC taxonomy for several case study queries. (**A**) "lung cancer" and "cisplatin" (data shown as Raw counts), (**B**) Colorectal cancer and Aspirin (data shown as CPROP; conditional probability), (**C**) growth factors "EGF" and "VEGF" (data shown as NPMI; normalised pointwise mutual information), and (**D**) housekeeping genes "GAPDH" and "TBP" (data shown as NPMI). Each bar represents the association for a cancer hallmark and/or associated biological process with the search query. The p-value is based on either Fisher-exact test, or a χ^2 test followed by a Bonferroni correction.

5.2 The LION system for literature-based discovery

The motivation behind literature-based discovery (LBD) is to support experts by semiautomatically discovering new knowledge from existing literature by connecting disjoint sets of literature that have not been directly connected previously (Kostoff, 2008).

LBD systems are becoming increasingly relevant and necessary, as the volume of scientific literature is growing at an exponential rate and it is notably time consuming and expensive to perform literature reviews. This makes researchers less likely to find pre-existing knowledge in literature, as well as generating new hypotheses by connecting disjoint sets of literature (Peng *et al.*, 2017).

LBD has led to many discovery proposals ranging from treatments for cataracts (Kostoff and Briggs, 2008), multiple sclerosis (Kostoff and Briggs, 2008), and Parkinson's disease (Kostoff and Briggs, 2008). There has also been numerous applications of LBD in the fields of drug development (Zhang *et al.*, 2014), drug repurposing (Yang *et al.*, 2017), and adverse drug event prediction (Hristovski *et al.*, 2016).

There are various models for LBD, for example, literature networks for knowledge discovery (Ivanisenko *et al.*, 2015). However, according to recent and comprehensive reviews of published biomedical LBD systems by Sam and McInnes (2017) and Sebastian *et al.* (2017), nearly all LBD systems are based on the ABC model of discovery (described in the next section). Therefore, within the context of this work we use "LBD" to refer to systems that are based on the ABC model.

In this section we describe LION LBD system, a system that is chiefly designed to support and automate the process of LBD (as well help with literature review in general) for cancer biology. The main target users of this system are cancer researchers, but also biomedical researchers in general as well as other cancer practitioners. The LION LBD system can be found at the following URL:

http://lbd.lionproject.net

The LION system is developed as part of an interdisciplinary collaboration with cancer researchers at Karolinska Institutet — Institute of Environmental Medicine [40] as well as researchers at Cancer Research UK Cambridge Institute (Cambridge University) — Narita Group [41]. Both of these groups have provided us with expert domain knowledge as well as evaluation data and regular user feedback during the development of the system.

In the proceeding sections, we will first describe relevant background information about the ABC model, then we detail the LION LBD system's functionality, implementation and evaluation.

[40] http://ki.se/en/orgid/82

[41] http://www.cruk.cam.ac.uk/research-groups/narita-group

5.2.1 The ABC model

Swanson (1987) made the earliest contributions to LBD research by using a combination of citation analysis and manual review for the discovery process. The key principle was to detect disjoint subsets of literature and then identify new connections across these sets. Swanson at the time only used titles of published biomedical text acquired from search results, but modern LBD system utilise other content from publications, typically abstract text and titles (Sebastian *et al.*, 2017).

The most well known example of Swanson's initial discoveries is the case of identifying a hidden connection between migraines and magnesium (Swanson, 1988). This was achieved by recognising several linking intermediate medical terms, such as "epilepsy" and "calcium channel blockers", that co-occurred frequently in the titles of both the magnesium literature set and the migraine literature set.

The main principle of Swanson's approach is identifying three categories of concepts: the initial concept (A, *e.g.* "migraine"), the target concept (C, *e.g.* "magnesium"), and the set of linking intermediate concepts (B, *e.g.* "epilepsy"). This is known as the ABC model, which makes the assumption that there is associative transitivity held between the concepts A and C *via* concept B. In other words, if "migraine" is correlated with "epilepsy", and "epilepsy" is correlated with "magnesium" then "migraine" is likewise correlated with "magnesium". A discovery hypothesis would then be made linking A and C which is new and does not previously appear in literature.

There are two kinds of discovery approaches in the ABC model:

- *Closed discovery*: The objective is to identify the most promising linking concepts (*i.e.* the terms that appear in B) that connect a provided starting concept (A) to another provided target concept (C). As shown in Figure 5.6. Closed discovery can be considered a method for hypothesis testing through literature (Sang *et al.*, 2015). For example, a researcher might already know that a given disease is treatable using a given substance, and has hypothesised the biological process that explains the effectiveness of the substance. The researcher could provide the disease as an input to node A and the substance to node B, and the system would generate the set of intermediate B nodes that connect the two concepts. The researcher can then test the hypothesised biological process by confirming the set of B nodes.

- *Open discovery*: The objective is to identify both the linking concepts (B) and target concepts (C) given only the initial concept (A). As shown in Figure 5.7. Open discovery is usually viewed as a process of hypothesis generation (Sang *et al.*, 2015). For example, a researcher could form the hypothesis that a disease represented by node A can be treated by a substance in a node C through a biological process represented by node B.

There have been several biomedical LBD systems developed over the years that are capable of conducting open discovery, closed discovery, or both. Table 5.4 summarises some notable examples.

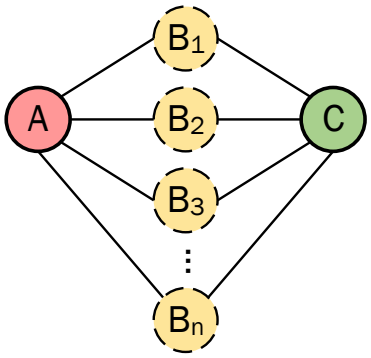


Figure 5.6: An illustration of closed discovery, where terms for concepts A and C are provided by the user (indicated by the bold outline), and the LBD system "discovers" intermediate (B) nodes that links A and C in literature (by co-occurrence).

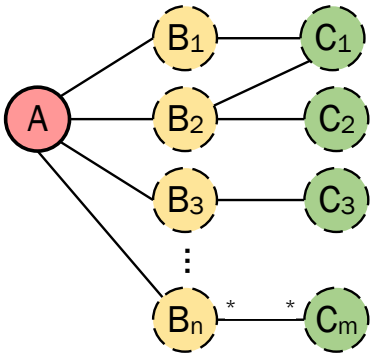


Figure 5.7: An illustration of open discovery, where only concept (A) is provided by the user, and the LBD system 'discovers' intermediate (B) nodes and the target (C) nodes and the connections between them.

5.2.2 Functionality overview

The LION LBD system is composed of four major components, as illustrated in Figure 5.8. The basic use case of the system is when a user begins to enter a search string in the UI (Step 1), which is then disambiguated and mapped to a unique concept identifier (Step 2). This is achieved *via* an autocomplete dropdown list in the UI which allows the user to select one of the mappings that contains the search string which points to a concept identifier (shown in Figure 5.9).

Once the concept is selected by the user, the concept graph is queried (Step 3) with concept identifier, and returns to the UI (and hence the user) a subgraph of the concept graph that is relevant to the user (illustrated as Step 4), *i.e.*, a graph for open discovery (Figure 5.3) or closed discovery (Figure 5.2). The user can continue to further query the concept graph (Step 5) by several means:

- By expanding the neighbouring nodes of the subgraph that is returned to the user.

Table 5.4: Currently active biomedical LBD systems that are based on the ABC model of discovery.

LBD System	Discovery type	Description
`Arrowsmith` (Swanson and Smalheiser, 1997)	Closed	The earliest closed discovery LBD systems, which was more recently updated (Smalheiser *et al.*, 2009). The system works by performing statistical analysis on co-occurrences of terms in `MEDLINE` article titles. http://arrowsmith.psych.uic.edu
`BITOLA` (Hristovski *et al.*, 2006)	Both	Based on `MEDLINE` citations, it uses co-occurrence relations of entities from UMLS and open biological ontologies (`OBO`), as well as relations extracted from `UMLS Semantic Network` and `SemRep`. http://ibmi.mf.uni-lj.si/sl/bitola
`FACTA+` (Tsuruoka *et al.*, 2011)	Both	An LBD system that incorporates both co-occurrence relationships as well as relationships labeled with biomedical events from the `genia` event corpus. http://www.nactem.ac.uk/facta/
`HiDE` (Preiss *et al.*, 2015)	Open	Extracts `UMLS` term co-occurrence relationships as well as syntactic relationships such as grammatical relation information, and subject-relation-object triples extracted using `SemRep` from the `PubMed` corpus. http://skye.shef.ac.uk/kdisc/hide-more

- By selecting another weight metric (*i.e.* selecting another way to measure co-occurrence between the concepts), There are currently nine metric types supported.
- Restricting the concept graph to a time period (by providing a maximum and minimum year filter for nodes and edges).
- Filtering the graph by concept type, *i.e.* displaying only some of the six concept types currently supported by LION: diseases, chemicals/drugs, genes/proteins, species, mutations, and cancer hallmarks.
- Restricting the graph by metric weight, where a maximum and minimum weights can be provided (for example, to filter out rare edges and nodes with low mention counts).

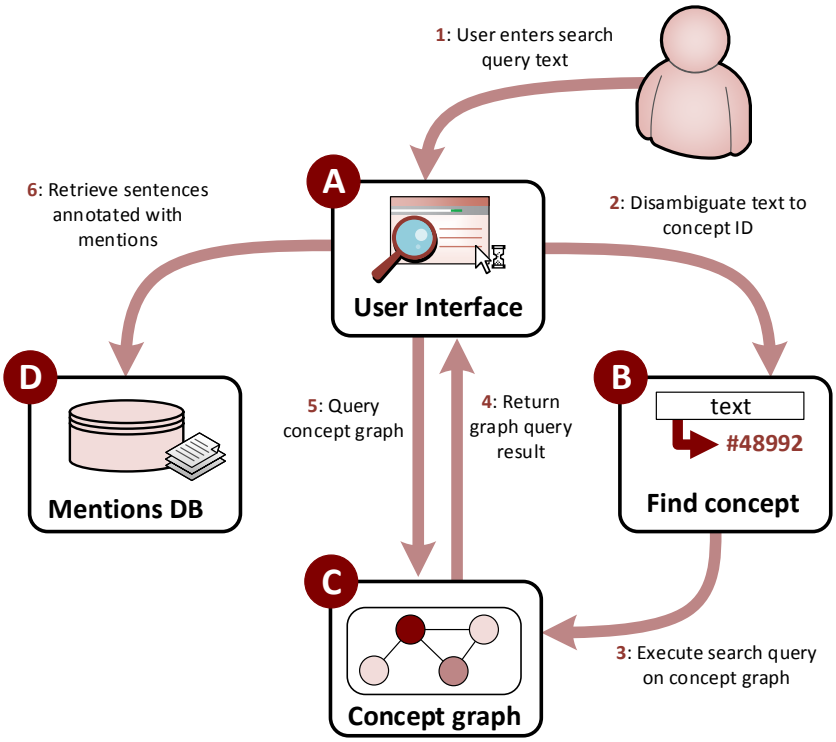


Figure 5.8: An illustration of the basic use case of the LION LBD system and the interaction between the system's main components.

Finally, in Step 5, the user can click an edge in the graph to view evidence from the text of the co-occurrences represented by the given edge in the concept graph (as shown in Figure 5.12). This queries the mentions database which stores all of the mentions between every concept represented by the concept graph.

In addition to presenting the concepts and relations to the user as a graph, the LION tool's UI can also present the result queried from the concept graph using a conventional IR style ranked search results paradigm (shown in Figure 5.13). This can be useful when the concept graph is too large or dense for the user to analyse and navigate.

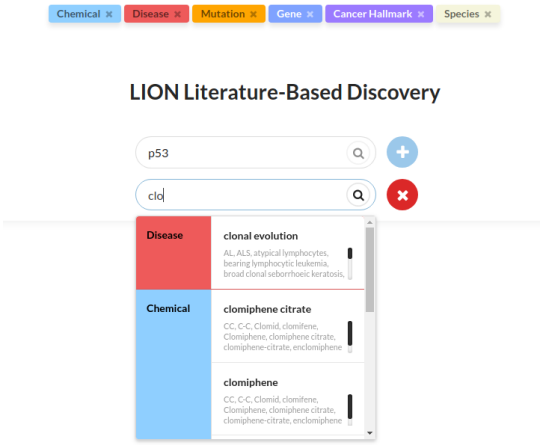


Figure 5.9: The user inputs a search string for open discovery and optionally a second string for closed discovery. The autocomplete dropdown list allows the user to disambiguate the search string by selecting a concept (listed by its canonical name).

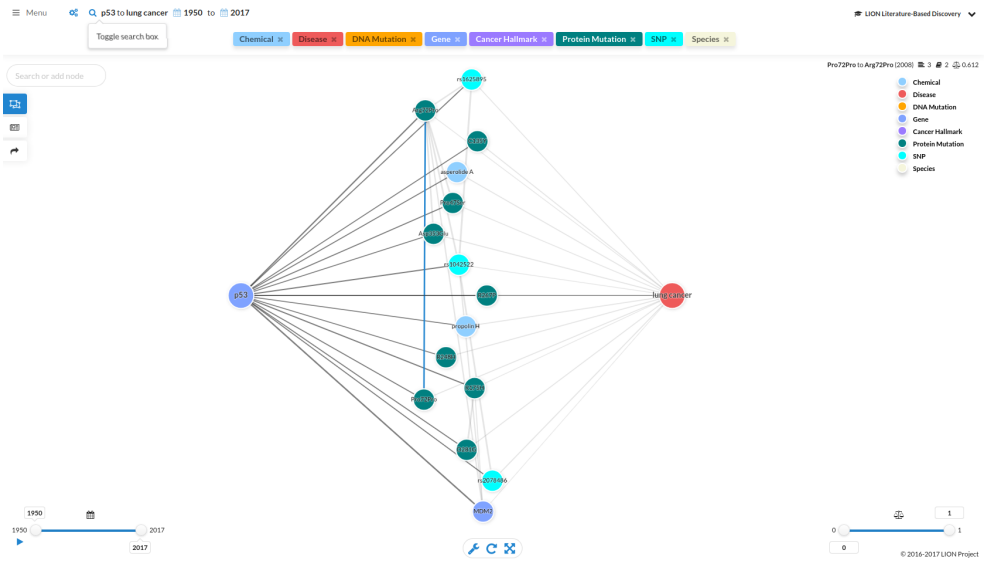


Figure 5.10: A screen image displaying the concept graph result of a closed discovery search. The nodes are grouped together where the intermediate (B) nodes are in the middle.

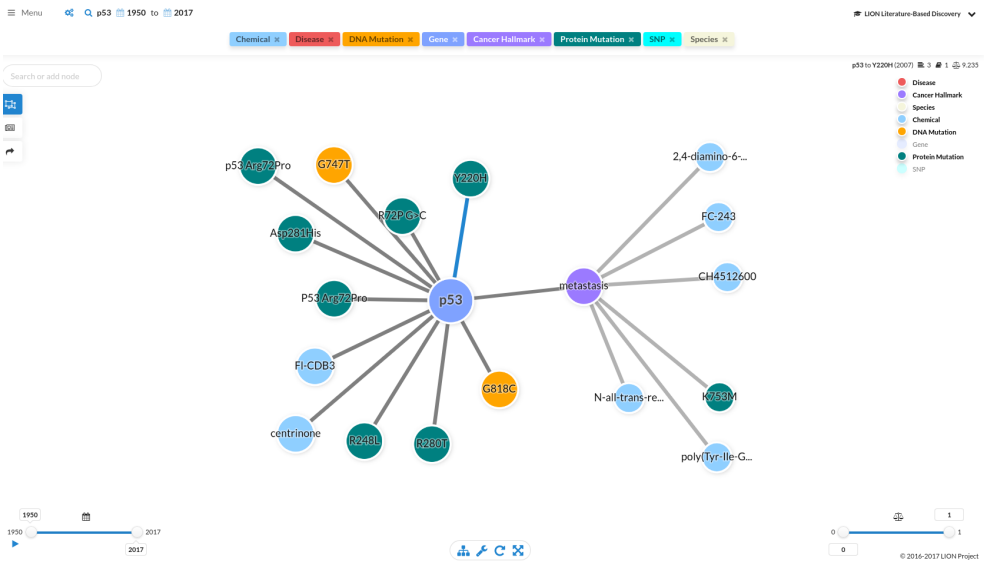


Figure 5.11: A screen image displaying the concept subgraph result of an open discovery search. A user can expand nodes to reveal further connections between concepts.

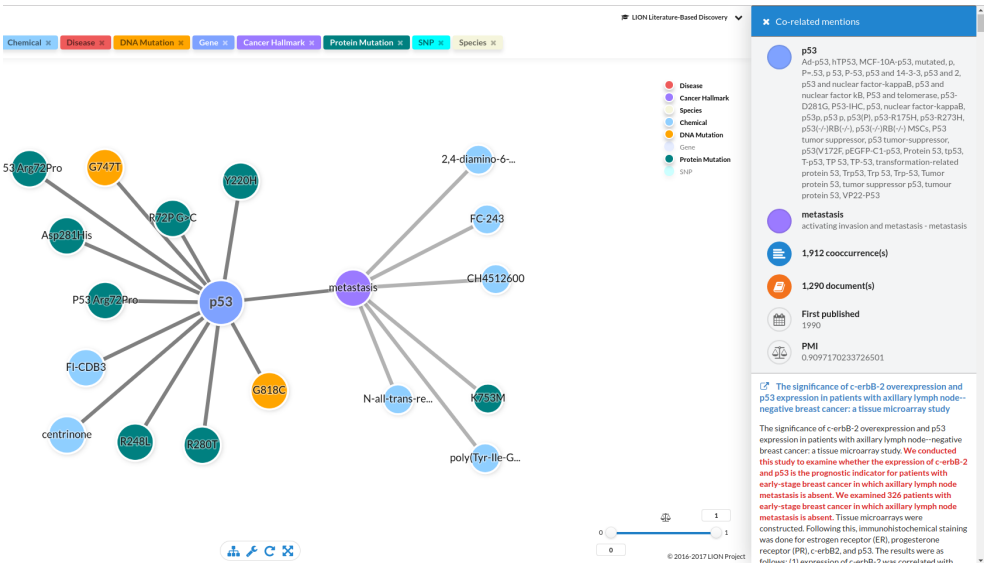


Figure 5.12: Selecting an edge on the concept graph would reveal a side panel displaying mention-level co-occurrences as supporting evidence. The user can also open a link to the `PubMed` article that contains the co-occurrence.

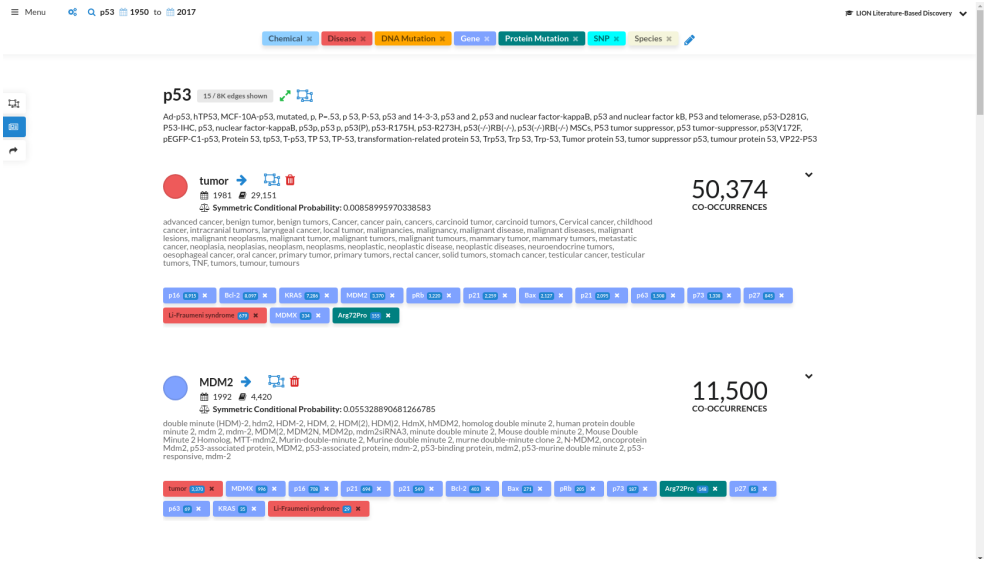


Figure 5.13: LION provides the option of switching between the concept graph display mode and more conventional search text-based results. This can be useful if the concept graph search result is too large and dense to analyse.

5.2.3 Co-occurrence metrics

One of the core objectives of any LBD system is to suggest useful discovery and to filter out noise. This is typically achieved by ranking results according to some metric or criteria (Yetisgen-Yildiz and Pratt, 2006).

In this section we describe a set of metrics that we use in the LION tool for ranking suggestions for both open and closed discovery modes. We utilise standard co-occurrence based metrics that are widely used in NLP, and are described as a staple of NLP literature (Jurafsky, 2000; Manning *et al.*, 1999).

We refer to Figure 5.14 as a scheme for the variable names in the equations that follows. The parametrisation used in the LION system involve four variables: the number of concept mentioned for each of the two connected nodes (a and b), the number of co-occurrences of the two concept (*i.e.* number of sentences that the two concepts co-occur) represented as j, and finally the total number of sentences in the corpus (n).

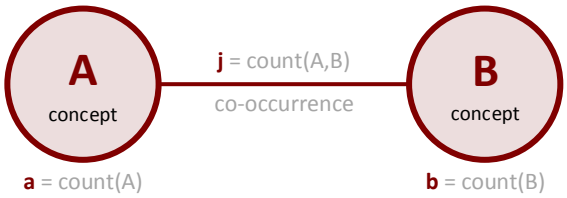


Figure 5.14: The LION LBD concept graph showing mention and co-occurrence counts.

The most basic metric used in LION is the absolute sentence-level co-occurrence count (C) which is simply equal to j. Similarly, we use number of documents (abstracts) that contain sentence-level occurrence as another metric (D). We also use PMI and NPMI described in Section 5.1.1 and restate as follows:

$$\text{PMI} = \log\left(\frac{P(A,B)}{P(A)\cdot P(B)}\right) = \log\left(\frac{jn}{ab}\right) \tag{5.4}$$

$$\text{NPMI} = \frac{\text{PMI}}{-\log\left(P(A,B)\right)} = \frac{\log\left(\frac{jn}{ab}\right)}{\log\left(\frac{n}{j}\right)} \tag{5.5}$$

We use symmetric conditional probability (SCP), which measures the ratio of the probability of the co-occurrence to the expected probability of the two concepts co-occurring independently. It has a value between one and zero, with one indicating perfect correlation between the two concepts and zero if they never appear together.

A value of more than 0.5 implies the two concept co-occur more frequently than expected. SCP is calculated as follows:

$$\text{SCP} = \frac{P(A,B)^2}{P(A) \cdot P(B)} = \frac{j^2}{ab} \tag{5.6}$$

Jaccard similarity J measures the ratio between the intersection of two sets to their union. It is calculated as follows:

$$\text{J} = \frac{j}{a+b-j} \tag{5.7}$$

Unlike the other metrics described here (with the exception of C and D), SCP and J do not require the number of sentences n, and are therefore only dependant on the relative proportions of the co-occurrences and mention counts of the given concepts.

We also use metrics based on hypothesis testing statistics. We use the χ^2 statistic which is calculated as follows:

$$\chi^2 = \sum_{x,y} \frac{O_{x,y} - E_{x,y}}{E_{x,y}} = \frac{n\left(j\left(n+j-a-b\right) - \left(a-j\right)\left(b-j\right)\right)^2}{ab\left(n-a\right)\left(n-b\right)} \tag{5.8}$$

where $O_{x,y}$ and $E_{x,y}$ are the observed and expected values of each cell in contingency matrix. We also calculate the Student's t statistic as follows:

$$t = \frac{\bar{x} - \mu}{\sqrt{\frac{\sigma^2}{n}}} \approx \frac{nj - ab}{n\sqrt{j}} \tag{5.9}$$

where $\bar{x}$ is the sample mean, μ is the population mean, and σ^2 is the variance.

Lastly, we use the log likelihood ratio (LLR) as a metric in LION. This statistic tests whether a sample is from a set of data with a specific distribution described by a hypothesised model (McInnes, 2004). LLR was first applied by Dunning (1993) as a way to determine if the words in an observed n-gram come from a sample that is independently distributed. LLR is defined as follows:

$$\text{LLR} = -2\log\lambda = \log\frac{\mathcal{L}\left(P(B \mid A) = P(B)\right)}{\mathcal{L}\left(P(B \mid A) = P(A,B \mid A)\right)} \tag{5.10}$$

where λ is the likelihood ratio, and $\mathcal{L}$ is the likelihood function. For our parametrisation, LLR can be estimated as follows:

$$\begin{aligned}\text{LLR} \approx 2\Bigg(j\log\left(\frac{jn}{ab}\right) + (b-j)\log\left(\frac{n(b-j)}{b(n-a)}\right) + (a-j)\log\left(\frac{n(a-j)}{a(n-b)}\right) \\ + (n-a-b+j)\log\left(\frac{n(n-a-b+j)}{(n-a)(n-b)}\right)\Bigg)\end{aligned} \tag{5.11}$$

For both open and closed modes of discovery ranking in the ABC model, there needs to be a method for combining the metric measures for pairs of edges (A–B and B–C), since the system would need to rank the combination of both edges in order to make a discovery (*i.e. via* Swanson chaining). For the work presented here, we will investigate three ways of aggregating the pairs of edges: the mean-average, the minimum, and the maximum value of edge metrics.

5.2.4 Implementation

The most visible component of the LION LBD system to the user is the UI which is a thin client implemented mostly in `JavaScript` and several libraries (mainly `Semantic UI`[42] and `Cytoscape.js`[43]). The second component is a mapping from names (mentions) of concepts extracted by our system, to the concept identifier that uniquely points to a concept in the concept graph.

Within the context of this dissertation, the system's backend (*i.e.* the concept graph and the mentions database) is perhaps the most relevant. Figure 5.15 shows a summary illustration of the key steps for constructing the systems backend.

There are five sources of information used to construct the LION system:

A `PubMed`: The basic text resource is the `PubMed` corpus containing 26,895,214 article abstracts and title citations. We also use metadata such as publication year.

B Hallmarks of Cancer: The full HoC taxonomy and corpus described in (Section 3.2.1) is used for identifying mentions of the hallmarks in the text as one of the six entity types used in the system. The hallmarks mentions are identified as sentences, unlike the other five entity types, where their mentions are typically single words or a short noun phrase identified using an NER system.

C `PubTator`[44]: All of the `PubMed` text used by LION is annotated with five concept types (*i.e.* named entity): diseases, chemicals/drugs, gene/protein,

[42] https://semantic-ui.com/
[43] http://js.cytoscape.org/
[44] http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/PubTator/

species, and mutations (which include DNA mutations, protein mutations, and single-nucleotide polymorphism mutations).

D `UniProt`[45]: The gene/protein entities identified by `PubTator` uses the `NCBI Gene`[46,47] identifiers. We also use `UniProt` (an open database of protein sequences) as an intermediate database to communicate with `PRO` (See step 1 of this pipeline).

E `Protein Ontology`[48] (`PRO`): The proteins identified by `PubTator` also encode orthologous[49] genes/protein information for different species. For example "p53" has several variants for humans, mice, *etc.*. After consulting cancer researchers at both Karloniska Instutet and the MRC Cancer Unit in Cambridge, we were advised that orthologues are typically not used in cancer research, and that having such information would introduce noise into the LION LBD system. Therefore, we use `Protein Ontology` that shows hypernymy relationships between orthologous proteins and their cross-species equivalent to map multiple orthologues of the same protein to a single protein (*e.g.* p53 in the previous example).

The pipeline uses the following steps to construct the LION LBD system which results in building two fundamental artefacts of the system's backend, the concept graph, and the mentions database.

1. The objective of this step is to merge mentions of entities that are orthologues of the same gene/protein. The motivation for this is that cancer researchers typically use cross-species analysis of sequence and gene interactions ignoring orthologues of the same gene/protein (Lu *et al.*, 2011). This step also helps reduce some noise in the data when merging several orthologous genes/proteins into a single entity. We achieve this merger by first converting the `NCBI Gene` identifiers into `UniProt` identifiers as an intermediary identifier that is then converted to `Protein Ontology` identifier. `Protein Ontology` stores orthologous (species-specific) proteins and the cross-species version of the same proteins in a hypernymy relationship, where the orthologous protein is a subnode. This provides a many-to-one mapping that allows us to merge orthologous proteins.

2. In this step we extract metainformation from `PubMed` for each article in our annotated corpus. The key information extracted in this step is the publication year which enables us to evaluate our system using the time slicing method,

[45] http://www.uniprot.org/

[46] https://www.ncbi.nlm.nih.gov/gene

[47] Previously known as `Entrez Gene`

[48] http://pir.georgetown.edu/pro/

[49] Orthologues are genes in different species that evolved from a common ancestral gene. Orthologues typically retain the same function in the course of evolution into different species (Koonin, 2005).

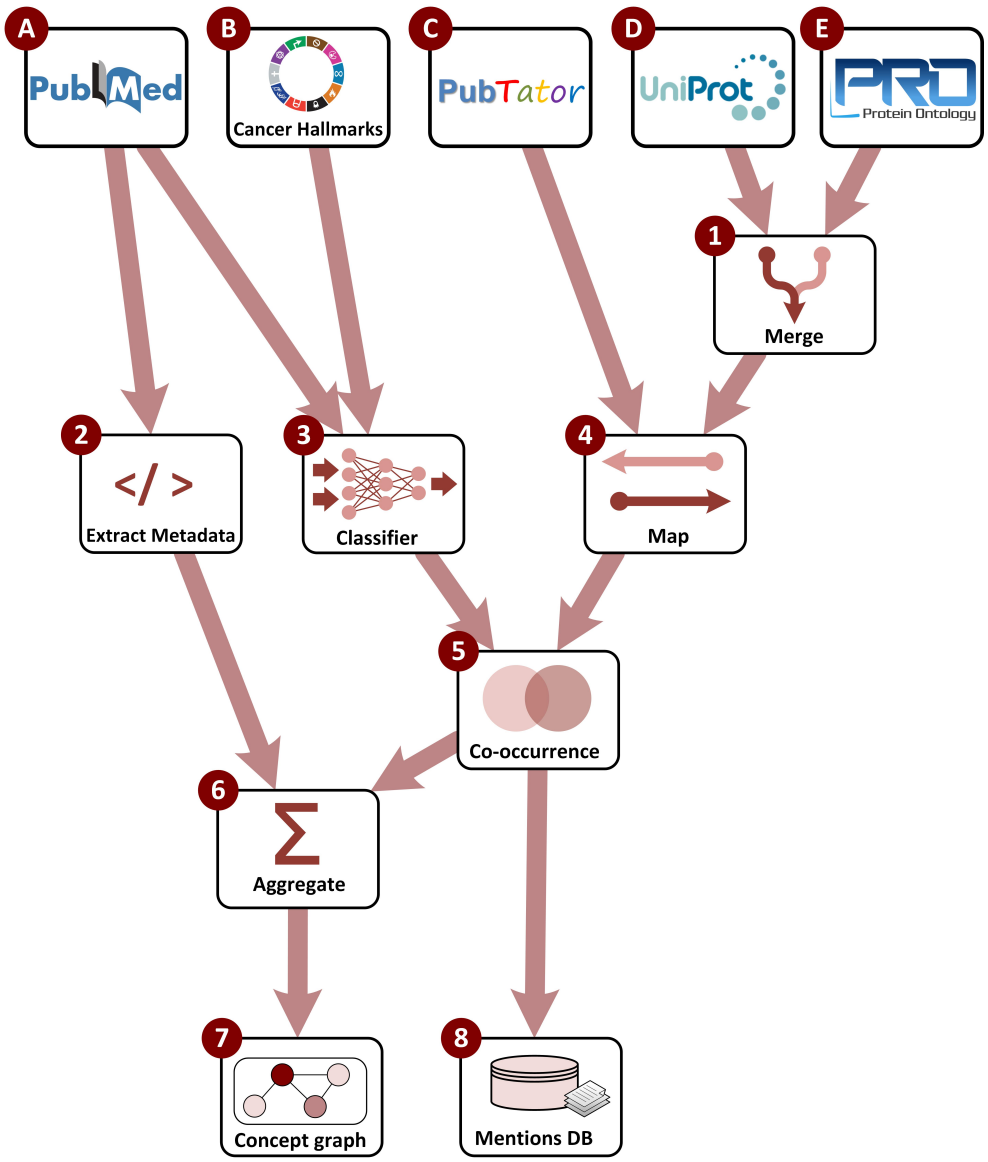


Figure 5.15: An illustration the main construction pipeline for the LION LBD system.[50]

[50] The author mostly focused on implementing components B, 3, 6, and 7, but also contributed to ideas relating to other components of the LION LBD system.

but also to have an insight on how nodes (concepts) and edges (co-occurrences) are distributed over time.

3. We train a CNN classifier using the HoC corpus. We use an identical architecture for the CNN classifier as that described in Section 4.3.3 for model INIT-B. The CNN model is then used to label each sentence in the `PubMed` corpus with one or more label in the HoC taxonomy.

4. The named entity mentions obtained from `PubTator` (Step C) and the merged identification acquired from `Protein Ontology` (Step 1) are combined into a single annotated corpus. This step also involves selecting the *canonical names*: the textual description of each concept that would be used as the descriptive text for the nodes on the concept graph (and likewise the UI). The canonical names are obtained from several sources. For example, if some entity mentions are identified with MeSH or other NCBI resources, then their canonical names are obtained from those ontologies or taxonomies; otherwise, the most common mention for a given concept would be selected as its canonical name.

5. The co-occurrence of entity mentions is extracted from the text within the span of one sentence. If more than one mention of the same entity appears per sentence then it is counted as a single occurrence, for example: "Two **patients** with a from **anastomotic leakage**); therefore, **anastomotic leakage** rate was 4%."; there are two highlighted NEs "patients" (*i.e.* human), and "anastomotic leakage"; although the latter co-occurs twice with the first within the span of this sentence, it is only counted as a single co-occurrence. Similarly, since the hallmarks annotation spans the entire sentence, co-occurrences are counted only once if the co-occurring concept appears more than once in the sentence.

6. All of the mentions and co-occurrences extracted in previous steps are counted and stored, as well as the time information extracted from `PubMed` (Step 2). This is achieved by compiling two lists.

   The first list contains concepts, where we store a unique concept identifier (which is the ontology identification previously tagged to mentions by an NER tool (`PubTator`), or the merged identifiers from `Protein Ontology`, or identifiers assigned from the HoC taxonomy), the counts of mentions for each concept and the earliest publication year of the concept, *i.e.*, the year of the publication that contains the first mention of the given concept.

   The second list stores the co-occurrences, which contains the IDs of the co-occurring concepts, the total number of co-occurrences of mentions between every pair of concepts that appear together (at sentence level), and the earliest publication year where such co-occurrence was observed.

   The aggregation step generates 450,671 concepts and 15,551,662 co-occurrences aggregated from 169,266,877 sentences.

7. The aggregated mention and co-occurrence information is transformed into a concept graph. The graph is stored and represented using the graph database `Neo4j`[51]. The nodes in the graph are concepts while edges represent co-occurrences between these concepts. Nodes in the graph store the concept's identifier, canonical name, total mention count for the concept, the earliest publication year, and the total document count that the concept is mentioned in. The edges on the graph store the counts of the co-occurrences of the connecting concepts, the earliest publication year of the co-occurrence, and precomputed values for all metrics described in Section 5.2.3.

   Edges in the graph that represent a single co-occurrence count (*hapax legomenon*) are excluded prior to constructing the graph along with nodes that represent a single mention. This filtering reduces some of the noise generated by errors earlier in the pipeline as well as greatly reducing the computational demands of the system. After *hapax legomenon* filtering, the graph consists of 254,818 nodes and 7,233,539 edges.

8. The mentions database contains all concept mentions identified in `PubMed` (459,623,741) and are searchable given the concept's unique identifier. The mentions database also stores all extracted co-occurrences from the data (258,309,875). The mentions database is implemented using `PostgreSQL`[52].

5.2.5 Evaluation

In this section we present a preliminary evaluation of the LION LBD system. As the system is under development at the time of conducting these experiments, all performance results and figures presented in this section are subject to change.

All experimental results outlined in this section were conducted on a restricted subset of `PubMed`, namely, all abstracts that have been tagged with the MeSH tag "neoplasm". The reason for restricting the `PubMed` corpus is mostly due to computational constraints: the goal here to assess a large set system parameters, which would otherwise not be possible to compute within the project's schedule. We therefore selected most cancer relevant subset of the `PubMed` corpus for this evaluation.

The literature restricted corpus for this evaluation would result in smaller concept graph that contains 60,303 nodes and 1,509,581 edges, which is 23.7 % of the nodes and 20.9% of the edges in the full `PubMed` scale concept graph.

The goal of the this evaluation is to measure the performance of all nine metrics supported by LION and likewise the use of the metrics in conjunction with the three aggregation functions (AVG (mean), MIN, and MAX).

We first measure how well our metrics correlate with one another in our system. We do so by selecting 100 edges at random, and measure the Spearman's rank correlation scores ρ. This metric measures how well two variables (in our case

[51] https://neo4j.com/
[52] https://www.postgresql.org/

Table 5.5: A pairwise comparison using Spearman's rank correlation (ρ) of all co-occurrence metrics used by LION LBD.

	C	D	PMI	NPMI	SCP	J	χ^2	t
D	0.811***							
PMI	−0.127	−0.232						
NPMI	−0.084	−0.194	0.997***					
SCP	0.238	0.094	0.894***	0.910***				
J	0.151	0.091	0.693***	0.707***	0.751***			
χ^2	0.141	0.020	0.865***	0.879***	0.916***	0.712***		
t	0.444**	0.261	0.685***	0.717***	0.796***	0.582***	0.713***	
LLR	0.370*	0.235	0.658***	0.684***	0.805***	0.620***	0.931***	0.709***

*** : $p < 0.001$; ** : $p < 0.01$; * : $p < 0.05$

metrics) monotonically correlate. We repeat the sampling process 1000 times and average the Spearman's scores. We do this for every pair of metric types as shown in Table 5.5.

The Spearman's rank test results show that there are two groups of metrics that correlate well with each other: the first group are metrics that use only absolute counts (C and D), and the second group are all other 'derived' and more sophisticated metrics described earlier. Document co-occurence counts (D) only correlates with absolute co-occurrence counts (C) and nothing else, while Student's t and LLR are the only two metrics that correlate with all other metrics. This implies that absolute count metrics would result different ordering compared to derived metrics on average.

The second experiment aims to ascertain how these metrics and aggregation functions perform on our concept graph under an experimental setup that aims to emulate open discovery. We use an established experimental protocol for evaluating LBD systems known as time slicing (or time travel) evaluation.

Under this setting, we organise and split our literature by time (we use publication year). We aggregate and construct the concept graph based on literature published strictly before a given cut off year, and then we try to predict direct co-occurrence relationships (*i.e.* A–C links) that appear in after the cut-off year, *i.e.*, we pretend that we have literature only from the past (before the cut-off year) in order to predict co-occurrence relationships in the future.

By observing the distribution of the edges in the concept grave with respect to time (first year in which a co-occurrence is mentioned), we have selected the year 2002 as a cutoff year as it represents about the half-way cutoff point.

We first identifiy all "discoverable" A–C edges after the cutoff year and compile a pairs of A–C tuples that would represent our "gold" data. We then query LION for open discovery, where we provide the A concept and LION generates all A–B–C chains reachable from the A concept before the cutoff year. We then measure Precision@k for $k = 1, 10, 100$ by comparing the ranked suggestions from LION with the gold data: Precision@1 would simply take the top-ranked suggested chain

Table 5.6: Time slicing evaluation of LION LBD. 100 A concepts are selected at random along with all discoverable edges after the cutoff year 2002. We evaluate on the LION system that is aggregated from pre-2002 data only. The system suggests and ranks the A–B–C chains according to all described metrics and aggregation functions. We compare with the post-2002 discoverable edges and measure the results using precision@k where $k = 1, 10, 100$. All figures are macro-averages (expressed as percentages) over the selected 100 A concepts. For compression purposes, a random baseline for this experiment whereby the system returns all possible rankings of the A–B–C chains equally likely has a macro-average precision of approximately 0.2% (regardless of the value of k).

	Precision@1			Precision@10			Precision@100		
	MIN	AVG	MAX	MIN	AVG	MAX	MIN	AVG	MAX
C	2.0	24.0	24.0	1.1	10.4	10.4	1	3.7	3.7
D	1.0	25.0	25.0	0.8	10.1	10.1	0.9	3.6	3.6
PMI	3.0	3.0	1.0	2.2	2.0	3.4	1.1	1.3	2.2
NPMI	3.0	5.0	11.0	2.7	2.7	5.7	1.1	1.5	2.5
SCP	3.0	17.0	18.0	2.7	7.0	7.7	1.5	3.4	3.4
J	2.0	18.0	19.0	2.4	7.2	8.0	1.4	3.5	3.8
χ^2	3.0	13.0	14.0	2.7	5.9	6.5	1.4	3.0	3.0
t	2.0	13.0	1.0	2.6	7.8	8.0	1.5	3.0	3.1
LLR	4.0	13.0	13.0	3.1	7.3	7.3	2.1	3.1	3.1

from the A–B–C chain, and for Precision@10 we take the top 10 ranked results and calculate precision with respect to the gold data. We randomly select 100 A concepts from our gold data for this experiment, and we average the results over those 100 runs. Table 5.6 shows the results of this experiment.

Overall, the system performs much better than what is expected using a random baseline which has a macro-averaged precision of 0.2%. We can see that the overall precision declines as we consider a larger k cut-off (from 1 to 100). As k increases, the value of precision@k would continue to drop until it reaches a lower limit equal to the proportion of positives in the returned results (*i.e.*when there is 100% recall), which is also equal to the random baseline precision score.

Simple counts (C and D) achieve the highest precision for all k cut-offs; Jaccard similarity J generally outperformed all other nontrivial metrics.

The MIN aggregation function performed the worst overall regardless of the metric, while both AVG and MAX are comparatively similar in performance.

The most important observation to note from the results of these experiments is that the more "sophisticated" metrics did not improve over simple absolute counts (C and D). This is perhaps surprising, as all of the statistical metrics and likewise most of the information based measures are normalised by some definition such that high absolute counts do not bias the metric (and therefore ranking) such that only the highest co-occurrence relations appear at the very top of the ranking; for example, "cancer" and "patient" (*i.e.* recognised as "human" on the concept graph), and thereby allowing for more "interesting" relations to appear at the top of the ranking.

Table 5.7: Landmark cancer discoveries that have been suggested by cancer researchers for evaluating LION under closed discovery setup.

Discovery #	Concept A	Concept B	Concept C	Earliest co-occurrence		
				A-B	B-C	A-C
1	p21 (Gene/Protein)	Proliferative signalling (Hallmark)	Leukaemia (Disease)	1985	1950	1990
2	KRAS (Gene/Protein)	Metastasis (Hallmark)	breast cancer (Disease)	1984	1948	1987
3	mTOR (Gene/Protein)	Proliferative signalling (Hallmark)	breast cancer (Disease)	1999	1948	2001
4	mTOR (Gene/Protein)	AKT (Gene/Protein)	Senescence (Hallmark)	2000	2005	2011

It is also important to note that time slicing experiment asks the system to predict future co-occurrence, and not actual discoveries. It could very well be the case that basic absolute counts are better at predicting future co-occurrence than the other metrics.

The second experiment aims to predict actual discoveries rather than future co-occurrences. Our cancer experts have identified four key "landmark" cancer discoveries that are well recognised in cancer research. These discoveries have been confirmed to be "discoverable" using LBD. In other words, they satisfy the definition of viability as candidates for LBD, where the first co-occurrence of the A and C concepts must appear after the first co-occurrence of A and B, and likewise B and C, in order for an LBD system to discover the A–C link *via* time travel to the past prior to the first appearance of the A–C co-occurrence in literature.

For this experiment, the LION system runs in closed discovery mode, where both concepts A and C are provided as input. Table 5.7 summarises these discovery cases.

In this experiment, we restrict the system to consider only A–B and B–C that have a co-occurrence that appears before the first A–C. The system will rank all B concepts according to a given combination of metric and aggregation function. Table 5.8 details the results for all of the four closed discovery cases.

The closed discovery evaluation results show that the system has generally performed exceptionally well. In all four cases, LION was able to suggest the correct intermediate B concept using at least one combination of metrics and aggregation function within the 90th percentile of the rankings. In three of the discovery cases, the correct B concepts were in the top five suggestions by the system. For the case of Discovery # 3, the correct B term was suggested as the first candidate using three of the metrics. In other words, the system suggested a previously unseen link in the literature used by the system, which is confirmed to be a biologically accurate link. These results are promising given that the LION system is still under development, and that only a fraction of `PubMed` was used in this evaluation.

Contrasting the results for the two experiments, the more sophisticated metrics proved to be far more useful than the first experiment. Absolute count metrics also

Table 5.8: Evaluation results of the four closed discovery cases (outlined in Table 5.7) The four subtables show all combinations of metrics and aggregate functions for each of the four discoveries. The first figure in each cell is the rank of the correct B (linking) concept as suggested by the system, the value of 1 indicates that the system predicted the correct B concept as the top rank. The figure in parentheses following the rank figure in each cell shows the percentile of the rank: a rank of 1 achieves the 100th percentile. A random baseline (with a uniform distribution) will have a percentile score of 50. The top scoring metric/aggregation function combination are highlighted in bold.

Discovery # 1 — B candidates: 61

	MIN	AVG	MAX
C	28 (55.7)	8 (88.5)	8 (88.5)
D	23 (63.9)	**7 (90.2)**	**7 (90.2)**
PMI	45 (27.9)	57 (8.2)	57 (8.2)
NPMI	49 (21.3)	56 (9.8)	57 (8.2)
SCP	46 (26.2)	36 (42.6)	34 (45.9)
J	49 (21.3)	27 (57.4)	26 (59.0)
χ^2	51 (18.0)	22 (65.6)	21 (67.2)
t	55 (11.5)	55 (11.5)	58 (6.6)
LLR	50 (19.7)	12 (82.0)	12 (82.0)

Discovery # 2 — B candidates: 78

	MIN	AVG	MAX
C	21 (74.4)	5 (94.9)	5 (94.9)
D	19 (76.9)	5 (94.9)	5 (94.9)
PMI	16 (80.8)	35 (56.4)	53 (33.3)
NPMI	16 (80.8)	18 (78.2)	22 (73.1)
SCP	41 (48.7)	5 (94.9)	5 (94.9)
J	58 (26.9)	**4 (96.2)**	**4 (96.2)**
χ^2	62 (21.8)	6 (93.6)	6 (93.6)
t	16 (80.8)	**4 (96.2)**	**4 (96.2)**
LLR	60 (24.4)	**4 (96.2)**	**4 (96.2)**

Discovery # 3 — B candidates: 27

	MIN	AVG	MAX
C	3 (92.6)	3 (92.6)	3 (92.6)
D	2 (96.3)	3 (92.6)	3 (92.6)
PMI	**1 (100.0)**	17 (40.7)	25 (11.1)
NPMI	**1 (100.0)**	9 (70.4)	20 (29.6)
SCP	5 (85.2)	4 (88.9)	4 (88.9)
J	19 (33.3)	2 (96.3)	2 (96.3)
χ^2	7 (77.8)	3 (92.6)	3 (92.6)
t	**1 (100.0)**	2 (96.3)	2 (96.3)
LLR	2 (96.3)	2 (96.3)	2 (96.3)

Discovery # 4 — B candidates: 213

	MIN	AVG	MAX
C	21 (90.6)	6 (97.7)	5 (98.1)
D	27 (87.8)	6 (97.7)	5 (98.1)
PMI	43 (80.3)	9 (96.2)	6 (97.7)
NPMI	38 (82.6)	5 (98.1)	**3 (99.1)**
SCP	18 (92.0)	**3 (99.1)**	**3 (99.1)**
J	7 (97.2)	4 (98.6)	4 (98.6)
χ^2	28 (87.3)	**3 (99.1)**	**3 (99.1)**
t	11 (95.3)	**3 (99.1)**	4 (98.6)
LLR	29 (86.9)	4 (98.6)	4 (98.6)

performed well, where metric D attained the best result in the first discovery case); however, in three of the four discovery cases, the more sophisticated metrics such as Student's t outperformed absolute counts.

5.2.6 Discussion

There are many association metrics for co-occurrence that we have not explored. Likewise, there are also many other aggregation functions and ways to combine these metrics. It is possible that there are other combinations that could yield better performance results. Deciding on what combination of metrics and aggregation function to best recommend to the user remains an open research question which we plan to investigate further.

The experiments described in previous section have shown that co-occurrence counts are better predictors for future co-occurrence than more sophisticated metrics, but perhaps not always as good at predicting a biologically accurate relationship between concepts. More sophisticated metrics such as χ^2, Student's t, and PMI seem to have performed well in the cancer landmark closed discoveries that were recommended by cancer researchers, but not do as well when predicting future co-occurrences compared to absolute counts. This is an important finding from the experiments presented here which relates to the limits of using co-occurrence based methods.There are significant drawbacks with using co-occurrence based relations in LBD that has been cited by many in the field (Preiss *et al.*, 2012):

1. Co-occurrence relations do not provide an explanation of how concepts are related—for example, they do not detect causality between concepts.

2. Important linguistic signals, such as negation and quantification, are not detected by co-occurrence based metrics.

3. Co-occurrence relationships are a crude method of filtering out noise from the system. Biases in the data will always exist: a high number of occurrences of popular concepts like "neoplasm" or "p53" would significantly outnumber other concepts that might lead to real discoveries, penalising relationship weights connecting to popular concepts. Using measures such as PMI comes at the cost of introducing noise by giving higher scores for less frequent concepts, such as rare mutations.

There are many ways to improve the performance of LION. One obvious improvement is by introducing additional steps for the normalisation of concepts. For example, some concepts are duplicated in our concept graph *e.g.*"metastasis", which appears as a disease and so does "metastatic disease" along with several other variants, this is also further complicated by having a "metastasis" hallmark concept, these and many other concept duplications can confuse the user, and add further complexity to the system.

The system could also possibly be improved by training a classifier to perform closed and open discovery. For example a classifier can be trained to predict whether an A-B, B-C combination can discover a future A-C. The training data for this classifier can be constructed from unseen future A-C edges, *i.e.*if there is an A-B-C chain that can predict a future A-C link, it would be labelled as a positive training example, otherwise negative. This kind of classifier would obviously learn to predict future co-occurrence, and not necessarily biologically useful relationships as discussed previously, but this could still improve on the current system.

In addition to these approaches, future iterations of the system will include new relationship types acquired from other sources in addition to co-occurrence. For example, relationships extracted from ontologies (such as hypernymy relations), and gene regulation relationships can be acquired from gene banks. This would

could potentially add another level of complexity and new ways of introducing unwanted noise to the system, so further research will undoubtedly be required.

Lastly, we can improve the system by including deeper linguistic analysis such as dependency parsing, relation extraction, coreference resolution, negation, and word-sense disambiguation (WSD). Work by Preiss *et al.* (2015) has suggested that the use of linguistic-based relations improves accuracy of LBD by generating fewer hypotheses without disadvantaging coverage. It was also shown that LBD systems can be sensitive the effects word senses and can possibly benefit from including WSD in the NLP pipeline (Preiss and Stevenson, 2016).

5.3 Chapter summary

We described in this chapter real-world applications of methods covered in the previous two chapters. We first applied our methodology from Chapter 3 in building cancer hallmark analytics tool (CHAT): we adapted our NLP pipeline to classify sentences instead of documents, and measured co-occurrences between the labelled `PubMed` abstracts corpus and an input search query. We evaluated CHAT intrinsically and by conducting several case studies. Our results show that the methodology behind the CHAT classifier is reasonably accurate given the challenges of sentence level classification, and that CHAT provides useful application for cancer hallmark TM for researchers.

We also apply our methods from Chapter 4 in the new LION LBD tool, which operates using the ABC models of open and closed discovery. The LION system identifies several concepts (NEs) in `PubMed` text: diseases, species, genes/proteins, mutations, chemicals/drugs, and cancer hallmarks. We apply our CNN model with hierarchical initialisation for identifying the cancer hallmarks.

Although the LION LBD system is still under development, preliminary evaluation indicates that this line of research is moving in a promising direction.

Chapter 6

Conclusions

Cancer is a multidisciplinary topic, and as such, cancer literature can span multiple sub-domains. It is therefore important to be able to extract information from literature that appeals to a wide range of experts.

A fundamental component of a modern TM system is an accurate classifier. This typically requires training a supervised algorithm on large number of labelled examples, which is often difficult to acquire in biomedicine due to requirement for large amounts of effort from highly specialised experts. To mitigate this, we engineer features that can better capture correlations between the raw input text and the target labels in order to better leverage the relatively small amounts of expert annotated data. The feature engineering process as well as extraction of these features from text has several drawbacks. The features may be time-consuming to handcraft, computationally expensive to extract from text, and could require other dependencies in addition to the raw input text (*i.e.* a set of tools forming an NLP pipeline to extract the features). Additionally, domain experts may still be required to design these features and methods of acquiring them. Furthermore, extensive feature engineering makes it difficult for the classification NLP pipeline to be portable across domains (*i.e.* cope with domain variation).

Although classifier performance is critical in a modern TM or IE system, it is also important to assess another important aspect: user-based evaluation. This is important as intrinsic evaluation is conducted under a controlled setting, *i.e.*, it does not ascertain whether the task at hand is actually *useful* for researchers or practitioners—it identifies merely the classification accuracy of the system. In reality the system might be overfitting to these settings, or perhaps might fail to scale to real world data. It is therefore important to conduct user-based evaluation in addition to intrinsic evaluation.

6.1 Key contributions

In this section, we describe how the key five objectives of this work (given in Chapter 1 and summarised below) have been addressed.

1. Propose a semantic text classification task that is relevant to as many disciplines (domains) of cancer, and have wide range of applications and usage.
2. Develop techniques for constructing classifiers that can be trained on a small and sparse datasets.
3. Develop a techniques for building classifiers that mitigate the problem of domain variation.
4. Reduce the burden of feature engineering for semantic text classification.
5. Evaluate our techniques and methods with end-users—domain experts who can verify the utility of our approach.

Chapter 3 introduced two semantic text classification tasks: the Hallmarks of Cancer, where the aim is to classify text according to a taxonomy based on established and widely recognised traits and principles that describe the inception and proliferation of cancer, and the Exposure taxonomy, which classifies exposure assessment (a vital part of chemical risk assessment).

A taxonomy was developed for the HoC to expand on the original ten hallmarks of Hanahan and Weinberg (2000, 2011), allowing for more detailed class labels for text classification. A corpus of `PubMed` document abstracts was annotated at the sentence level according to these extended taxonomy class labels, allowing for both sentence and document (abstract) classification. A similar taxonomy and analogous corpora for sentence and document classification were developed for the task of exposure assessment.

A supervised, feature-rich NLP pipeline for classifying PubMed text was developed and applied to both the HoC and Exposure taxonomy tasks. The pipeline was evaluated on both tasks intrinsically, and attained good levels of performance when trained on our small and sparse annotated corpora. Several case studies were also conducted, whose results further support the usefulness of the classification by contrasting the results with real-world knowledge.

Chapter 4 focussed on the key challenge of reducing feature engineering while maintaining good accuracy performance of the classifiers on small and sparse datasets.

We proposed a method to jointly learn a multilevel embedded representation of the target class labels, as well as words, sentences, and documents in the same vector space. The intuition behind these multilevel embeddings is that each level captures slightly different topical semantics. These embeddings were employed to produce three types of features that can be extracted relatively efficiently, producing competitive results. We have also shown that porting a classifier trained using this approach from the general (newswire) domain to biomedical domain resulted in a smaller drop in performance, *i.e.*, less susceptibility to domain variation. The features were also shown to attain good performance with standard classification algorithms such as SVMs and with semisupervised classification where the decision boundaries are not learned from labelled data.

Next, neural models were used for automated feature learning. A CNN model taking only the input text and word representations induced from unannotated general domain text as input was demonstrated to achieve competitive performance on the Hallmarks of Cancer data compared to a feature-rich, manually-engineered SVM baseline.

The use of CNN and SVM classifiers for multilabel classification using a one-*vs*.-rest setup assumes a flat label structure. The drawbacks of this approach are the inability to leverage dependencies between classes in the training and classification process, and the additional computational cost of training a classifier for each class. Therefore, a method was implemented for multilabel classification that initialises a neural network model's final hidden layer to leverage label co-occurrence. This approach was evaluated using the HoC and Exposure taxonomies, both of which are hierarchical multilabel classification tasks. The experimental results demonstrated that overall, the method attained better results than any other model, with the sole exception of sentence-level classification of the Exposure taxonomy.

Chapter 5 discusses two real-world end-user applications (TM tools) that utilise the HoC task introduced in Chapter 3, as well as applying some of the methods developed in Chapters 3 and 4 to large-scale `PubMed` data.

We first introduced CHAT, a tool that enables users to see the co-occurrence of cancer hallmarks and any search term entered into the tool—for example, drug names, genes, chemicals, diseases, *etc*. The tool classifies every sentence in `PubMed` according to the HoC taxonomy, and then uses co-occurrence and probability and entropy metrics to measure the association between the hallmarks and the search term. CHAT utilised an NLP pipeline similar to the methodology in Chapter 3, but with the addition of the word distance feature that is extracted from the joint MLE space described in Chapter 4.

Evaluation of CHAT using standard intrinsic metrics showed good results, and its evaluation *via* case studies by cancer research scientists has reaffirmed existing domain knowledge.

The second tool is the LION LBD system: the first LBD system that specifically focuses on cancer research. The LION system is capable of conducting both open and closed discovery. LION utilises six concept types and co-occurrence relations extracted from more than 26 million `PubMed` abstracts (over 150 million sentences), and uses nine different co-occurrence metrics. Like CHAT, the LBD tool uses the HoC taxonomy to represent cancer processes. The hallmarks are one of its six entity concept types, which were extracted using the our hierarchical initialised CNN method as described in Chapter 4.

We conducted a preliminary evaluation of LION using time slicing as well as evaluating the system using four cancer "landmark" discoveries that are well known in cancer research under a closed discovery setup. We measured the performance of the nine metrics currently supported by LION, and the results have shown promising potential. This was particularly true for the landmark discoveries, where the system suggested the correct intermediate B concept in the top 90th percentile of rankings for all four case studies using at least one combination of metrics and

aggregate function; moreover, it predicted the correct answer for one discovery case as its first ranked candidate. The LION system is still under development, and further evaluation will be conducted in the near future.

6.2 Future directions

One potential future research direction is to utilise further information in our models. All of the methods used in this dissertation has been used exclusively on abstract text from PubMed. Although this has been shown to be useful (as abstracts typically summarise key findings and methods in a given publication), there are far more details that are not described. A natural progression of our work is to utilise full publication text from MEDLINE.[53] TM from full text articles has always been more challenging, where systems experienced a significant performance drop when migrating from abstract text to full-text publications due to either lexical differences in the text, or structural differences, *e.g.*, longer sentences and different writing styles (Huang and Lu, 2015). This is likely to also affect the current performance of our classifiers to some degree, but this has not yet been evaluated. It is likely that further research is required in order to adapt our classifiers to full text articles.

In addition to classifying full-text publications, another research direction is to combine other non-text content (such as images in figures) in these publications to aid in text classification. This integrates elegantly with our neural approaches, as these images would constitute an additional input signal into the network. Multimodal ANNs have witnessed wide popularity in recent years (Ngiam *et al.*, 2011) with broad applications such as image captioning (Vinyals *et al.*, 2015). It is also possible to apply these techniques to biomedical literature for several tasks in addition to text classification—for example, image caption generation or, more ambitiously, generating figures from text.

Likewise, our LBD system can be improved if additional information is also included. One possibility is to include data from other sources such as genomic databases—for example, the COSMIC database, which catalogues somatically-acquired mutations found in human cancer (Forbes *et al.*, 2010). Extra information can be included as additional relations to the system, or can be used to improve the current co-occurrence relations.

Another research direction to pursue is developing and adapting new ML techniques for biomedical text classification. The classification methods that have been described in this work are largely discriminative in nature—that is, they model the dependence of the target label on the input text, in contrast with generative models that model both the input and the target labels jointly. Recent developments in generative ANNs may prove beneficial for biomedical text classification. Recent methodologies involve simultaneously training a generative network and a discriminative network where the generative network is taught to map from a

[53] Full publication text is not always available; the open-access subset of MEDLINE is smaller but still of use.

latent space to a particular data distribution of interest, which is then sampled as input to the discriminative network. Some examples of ANN architectures that have recently gained prominence in the ML and NLP communities are generative adversarial networks (GANs) and variational autoencoders (VAEs). In the case of GANs, the discriminative network is trained to discriminate between instances from the true data distribution and synthesised instances produced by the generator—*i.e.*, the generative model attempts to "trick" the discriminator network by generating synthesised instances that appear to have come from the true data distribution (Goodfellow *et al.*, 2014).

The VAE is a similar architecture. It consists of an encoder (the discriminative network) and a decoder (the generative network); both components are trained jointly such that the VAE learns to predict its input. In the context of text classification, the VAE encoder can be used at inference time to classify unseen text (Xu *et al.*, 2017). For both of these architectures, using generative models allows us to leverage unlabelled data (which is abundant for biomedical text), along with a relatively small amount of labelled data for improved performance.

Finally, another direction of future work is researching further tools and applications so that they better meet the needs of end-users. Consultation with end-users of the two tools discussed in this thesis have revealed several avenues for future improvements. For example, CHAT could be further improved by distinguishing between positive and negative evidence for a particular hallmark, or between reported facts and speculations. For example, currently CHAT is incapable of detecting negations in text and therefore cannot correctly infer positive or negative association. We can include negation in several places in our NLP pipeline, for example as features in the classification process, or as an additional classification step. We can utilise resources such as `BioNØT` (a database of negated `PubMed` sentences) (Agarwal *et al.*, 2011) or the `BioScope` corpus (Szarvas *et al.*, 2008).

CHAT can be refined to consider journal impact factors, citation frequencies, and cross references. This could help cancer researchers to identify (for instance) more prominent, less important, and incremental published articles in the evidence screen. The tool can also be extended to support time-series analysis of the scientific data related to cancer, where it might be useful to see how the distribution of associated hallmarks changes with time for certain chemicals and drugs.

Likewise, the LION LBD tool can be further improved by including new types of relations between nodes in the graph in addition to co-occurrences, *e.g.*, relations to denote hypernymy between concepts, relations to denote causal relations like gene expression regulation, and protein binding, and relations expressing chemical–drug reactions and effects.

LION might also benefit from deeper linguistic analysis in both concept recognition, and co-occurrence extraction. For example, WSD, co-reference resolution, and negation detection can be included in the NLP pipeline for LION in addition to NER when identifying concepts in the text. Work by Preiss and Stevenson (2016); Preiss *et al.* (2012) has suggested that deeper linguistic analysis could aid in reducing noise and improving the performance of LBD systems.

Another avenue of research in LBD is *link prediction*: the problem of predicting the presence (or absence) of edges between nodes of a graph. Within the context of LBD, there are two types of link prediction: *structural* prediction, where the input is a concept graph and we wish to predict a relation between a pair of concepts, and *temporal* prediction, where we have a sequence of fully observed graphs at various time steps as input and our goal is to predict the graph at the next time step (Menon and Elkan, 2011). Both problems have important practical applications and are active areas of research. For example, link prediction has been applied to protein–protein interactions (Lei and Ruan, 2012), drug–drug interactions (Fokoue *et al.*, 2016), and adverse drug reactions (Lin *et al.*, 2013). Both types of link prediction can be applied to our LBD system, with different types of entities.

In conclusion, we believe that the contributions presented by this dissertation have been demonstrated to have practical real-world applications for cancer research, and a promising potential to aid cancer researchers and other practitioners interested in cancer text mining.

Bibliography

Abal, M., Andreu, J., and Barasoain, I. (2003). Taxanes: microtubule and centrosome targets, and cell cycle dependent mechanisms of action. *Current Cancer Drug Targets*, 3(3):193–203.

Agarwal, S. and Yu, H. (2009). Automatically classifying sentences in full-text biomedical articles into introduction, methods, results and discussion. *Bioinformatics*, 25(23):3174–3180.

Agarwal, S., Yu, H., and Kohane, I. (2011). BioNØT: A searchable database of biomedical negated sentences. *BMC Bioinformatics*, 12(1):420.

Aggarwal, C. C. and Zhai, C. (2012). A survey of text clustering algorithms. In *Mining Text Data*, pages 77–128. Springer.

Akinci, M., Aslan, S., Markoç, F., Cetin, B., and Cetin, A. (2008). Metastatic basal cell carcinoma. *Acta Chirurgica Belgica*, 108(2):269.

Arighi, C. N., Roberts, P. M., Agarwal, S., Bhattacharya, S., Cesareni, G., Chatr-Aryamontri, A., Clematide, S., Gaudet, P., Giglio, M. G., Harrow, I., *et al.* (2011). BioCreative III interactive task: an overview. *BMC Bioinformatics*, 12(8):S4.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., *et al.* (2000). Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25.

Asimakopoulos, A. G., Thomaidis, N. S., and Koupparis, M. A. (2012). Recent trends in biomonitoring of bisphenol a, 4-t-octylphenol, and 4-nonylphenol. *Toxicology Letters*, 210(2):141–154.

Baker, S., Ali, I., Silins, I., Pyysalo, S., Guo, Y., Högberg, J., Stenius, U., and Korhonen, A. (2017). Cancer hallmarks analytics tool (CHAT): A text mining approach to organise and evaluate scientific literature on cancer. *Bioinformatics*.

Baker, S., Kiela, D., and Korhonen, A. (2016a). Robust text classification for sparsely labelled data using multi-level embeddings. In *Proceedings of COLING 2016*, pages 2333–2343, Osaka, Japan.

Baker, S. and Korhonen, A. (2017). Initializing neural networks for hierarchical multi-label text classification. In *Proceedings of the 16th Workshop on Biomedical Natural Language Processing (BioNLP2017)*, Vancouver, Canada.

Baker, S., Korhonen, A., and Pyysalo, S. (2016b). Cancer hallmark text classification using convolutional neural networks. In *Proceedings of the Fifth Workshop on*

Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2016), pages 1–9, Osaka, Japan.

Baker, S., Silins, I., Guo, Y., Ali, I., Högberg, J., Stenius, U., and Korhonen, A. (2016c). Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440.

Barbosa-Silva, A., Fontaine, J.-F., Donnard, E. R., Stussi, F., Ortega, J. M., and Andrade-Navarro, M. A. (2011). PESCADOR, a web-based tool to assist text-mining of biointeractions extracted from PubMed queries. *BMC Bioinformatics*, 12(1):435.

Beers, M. H., Berkow, R., *et al.* (1999). The Merck manual. *Merck Research Laboratories*, 17.

Bergdahl, I. A. and Skerfving, S. (2008). Biomonitoring of lead exposure—alternatives to blood. *Journal of Toxicology and Environmental Health, Part A*, 71(18):1235–1243.

Bhatia, K., Jain, H., Kar, P., Varma, M., and Jain, P. (2015). Sparse local embeddings for extreme multi-label classification. In *Advances in Neural Information Processing Systems*, pages 730–738.

Bird, S. (2006). NLTK: The natural language toolkit. In *Proceedings of the COLING/ACL on Interactive Presentation Sessions*, COLING-ACL '06, pages 69–72, Stroudsburg, PA, USA. Association for Computational Linguistics.

Blaschke, C., Andrade, M. A., Ouzounis, C. A., and Valencia, A. (1999). Automatic extraction of biological information from scientific text: protein-protein interactions. In *ISMB*, volume 7, pages 60–67.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 144–152. ACM.

Burnside, B., Strasberg, H., and Rubin, D. (2000). Automated indexing of mammography reports using linear least squares fit. In *Proceedings of the 14th International Congress and Exhibition on Computer Assisted Radiology and Surgery*, pages 449–454.

Cancer Research UK (2017). Cancer survival statistics.

Cerri, R., Barros, R. C., and De Carvalho, A. C. (2014). Hierarchical multi-label classification using local neural networks. *Journal of Computer and System Sciences*, 80(1):39–56.

Chen, G., Ye, D., Xing, Z., Chen, J., and Cambria, E. (2017). Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2377–2383.

Chiu, B., Crichton, G., Korhonen, A., and Pyysalo, S. (2016). How to train good word embeddings for biomedical NLP. In *Proceedings of BioNLP*.

Chollet, F. (2015). Keras. https://github.com/fchollet/keras.

Clark, S. (2002). Supertagging for combinatory categorial grammar. In *Proceedings of the 6th International Workshop on Tree Adjoining Grammars and Related Frameworks (TAG+ 6)*, pages 19–24.

Clark, S., Hockenmaier, J., and Steedman, M. (2002). Building deep dependency structures with a wide-coverage ccg parser. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 327–334. Association for Computational Lingustics.

Clark, W. T. and Radivojac, P. (2013). Information-theoretic evaluation of predicted ontological annotations. *Bioinformatics*, 29(13):i53–i61.

Cohen, A. M. and Hersh, W. R. (2005). A survey of current work in biomedical text mining. *Briefings in Bioinformatics*, 6(1):57–71.

Cohen, K. B. and Demner-Fushman, D. (2014). *Biomedical natural language processing*, volume 11. John Benjamins Publishing Company.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Consortium, U. *et al.* (2014). UniProt: a hub for protein information. *Nucleic Acids Research*, page gku989.

CONTAM, E. (2009). Scientific opinion on arsenic in food. *EFSA Journal*, 7(10):1351.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Crichton, G., Pyysalo, S., Chiu, B., and Korhonen, A. (2017). A neural network multi-task learning approach to biomedical named entity recognition. *BMC Bioinformatics*, 18(1):368.

Dai, H.-J., Wu, J. C.-Y., Lin, W.-S., Reyes, A. J. F., Syed-Abdul, S., Tsai, R. T.-H., Hsu, W.-L., *et al.* (2014). LiverCancerMarkerRIF: a liver cancer biomarker interactive curation system combining text mining and expert annotations. *Database*, 2014.

Dai, W., Xue, G.-R., Yang, Q., and Yu, Y. (2007). Transferring naive bayes classifiers for text classification. In *Proceedings Of The National Conference On Artificial Intelligence*, volume 22, page 540. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 233–240. ACM.

Degtyarenko, K., De Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcántara, R., Darsow, M., Guedj, M., and Ashburner, M. (2007). ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic Acids Research*, 36(suppl_1):D344–D350.

Doland, M. E. (2014). Capturing cancer initiating events in OncoCL, a cancer cell ontology. In *AMIA Summits on Translational Science Proceedings*.

Drew, D. A., Cao, Y., and Chan, A. T. (2016). Aspirin and colorectal cancer: the promise of precision chemoprevention. *Nature Reviews Cancer*, 16(3):173.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Fang, Y.-C., Huang, H.-C., and Juan, H.-F. (2008). MeInfoText: associated gene methylation and cancer information from text mining. *BMC Bioinformatics*, 9(1):22.

FAO/WHO (1995). Application of risk analysis to food standards issues: report of the joint FA.

Federhen, S. (2011). The NCBI taxonomy database. *Nucleic Acids Research*, 40(D1):D136–D143.

Fidler, I. J. (1995). Melanoma metastasis. *Cancer Control*, 2(5):398–404.

Fleiss, J. L., Levin, B., and Paik, M. C. (2013). *Statistical methods for rates and proportions*. John Wiley & Sons.

Fleuren, W. W. and Alkema, W. (2015). Application of text mining in the biomedical domain. *Methods*, 74:97–106.

Fokoue, A., Sadoghi, M., Hassanzadeh, O., and Zhang, P. (2016). Predicting drug-drug interactions through large-scale similarity-based link prediction. In *International Semantic Web Conference*, pages 774–789. Springer.

Forbes, S. A., Bindal, N., Bamford, S., Cole, C., Kok, C. Y., Beare, D., Jia, M., Shepherd, R., Leung, K., Menzies, A., *et al.* (2010). COSMIC: mining complete cancer genomes in the catalogue of somatic mutations in cancer. *Nucleic Acids Research*, 39(suppl_1):D945–D950.

Frederiksen, H., Skakkebaek, N. E., and Andersson, A. M. (2007). Metabolism of phthalates in humans. *Molecular Nutrition & Food Research*, 51:899–911.

Fundel, K., Küffner, R., and Zimmer, R. (2006). RelEx—Relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365–371.

Gasull, M., de Basea, M. B., Puigdomènech, E., Pumarega, J., and Porta, M. (2011). Empirical analyses of the influence of diet on human concentrations of persistent organic pollutants: a systematic review of all studies conducted in spain. *Environment International*, 37(7):1226–1235.

Ginn, R., Pimpalkhute, P., Nikfarjam, A., Patki, A., O'Connor, K., Sarker, A., Smith, K., and Gonzalez, G. (2014). Mining Twitter for adverse drug reaction mentions: a corpus and classification benchmark. In *Proceedings of BioTxtM 2014*.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Aistats*, volume 9, pages 249–256.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680.

Goulart, R. R. V., de Lima, V. L. S., and Xavier, C. C. (2011). A systematic review of named entity recognition in biomedical texts. *Journal of the Brazilian Computer Society*, 17(2):103–116.

Govindarajan, P. and Ravichandran, K. (2016). Text mining from biomedical domain using a full parser. In *International Conference on Inventive Computation Technologies (ICICT)*, volume 2, pages 1–9. IEEE.

Greenhalgh, T. (1997). How to read a paper. the MEDLINE database. *British Medical Journal*, 315(7101):180.

Guo, Y., Korhonen, A., Liakata, M., Karolinska, I. S., Sun, L., and Stenius, U. (2010). Identifying the information structure of scientific abstracts: an investigation of three different schemes. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, pages 99–107. Association for Computational Linguistics.

Guo, Y., Korhonen, A., Silins, I., and Stenius, U. (2011). Weakly supervised learning of information structure of scientific abstracts–is it accurate enough to benefit real-world tasks in biomedicine? *Bioinformatics*, *27*(22):3179–3185.

Guo, Y., Ó'Séaghdha, D., Silins, I., Sun, L., Högberg, J., Stenius, U., and Korhonen, A. (2014). CRAB 2.0: A text mining tool for supporting literature review in chemical cancer risk assessment. In *COLING (Demos)*, pages 76–80.

Guo, Y., Silins, I., Reichart, R., and Korhonen, A. (2012). CRAB reader: A tool for analysis and visualization of argumentative zones in scientific literature. In *Proceedings of COLING 2012: Demonstration Papers*, pages 183–190.

Habibi, M., Weber, L., Neves, M., Wiegandt, D. L., and Leser, U. (2017). Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48.

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

Hanahan, D. and Weinberg, R. A. (2011). Hallmarks of cancer: the next generation. *Cell*, 144(5):646–674.

Hearst, M. A. (1999). Untangling text data mining. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 3–10. Association for Computational Linguistics.

Hersh, W. (2008). *Information retrieval: a health and biomedical perspective*. Springer Science & Business Media.

Hogenboom, F., Frasincar, F., Kaymak, U., De Jong, F., and Caron, E. (2016). A survey of event extraction methods from text for decision support systems. *Decision Support Systems*, 85:12–22.

Holzinger, A., Schantl, J., Schroettner, M., Seifert, C., and Verspoor, K. (2014). Biomedical text mining: state-of-the-art, open problems and future challenges. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*, pages 271–300. Springer.

Holzinger, A., Stocker, C., Ofner, B., Prohaska, G., Brabenetz, A., and Hofmann-Wellenhof, R. (2013). Combining HCI, natural language processing, and knowledge discovery-potential of IBM content analytics as an assistive technology in the biomedical field. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data*, pages 13–24. Springer.

Hristovski, D., Friedman, C., Rindflesch, T. C., and Peterlin, B. (2006). Exploiting semantic relations for literature-based discovery. In *AMIA Annual Symposium Proceedings*, volume 2006, page 349. American Medical Informatics Association.

Hristovski, D., Kastrin, A., Dinevski, D., Burgun, A., Žiberna, L., and Rindflesch, T. C. (2016). Using literature-based discovery to explain adverse drug effects. *Journal of Medical Systems*, 40(8):1–5.

Huang, C., Qiu, X., and Huang, X. (2014). Text classification with document embeddings. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 131–140. Springer.

Huang, C.-C. and Lu, Z. (2015). Community challenges in biomedical text mining over 10 years: success, failure and the future. *Briefings in Bioinformatics*, 17(1):132–144.

Hughes, M., Li, I., Kotoulas, S., and Suzumura, T. (2017). Medical text classification using convolutional neural networks. *arXiv preprint arXiv:1704.06841*.

Hunter, L. and Cohen, K. B. (2006). Biomedical language processing: what's beyond PubMed? *Molecular Cell*, 21(5):589–594.

Ivanisenko, V. A., Saik, O. V., Ivanisenko, N. V., Tiys, E. S., Ivanisenko, T. V., Demenkov, P. S., and Kolchanov, N. A. (2015). ANDSystem: an Associative Network Discovery System for automated literature mining in the field of biology. *BMC Systems Biology*, 9(2):S2.

Iyer, G., Wang, A. R., Brennan, S. R., Bourgeois, S., Armstrong, E., Shah, P., and Harari, P. M. (2017). Identification of stable housekeeping genes in response to ionizing radiation in cancer research. *Scientific Reports*, 7.

Jacobs, P. S. (2014). *Text-based intelligent systems: Current research and practice in information extraction and retrieval*. Psychology Press.

Janjua, N. R., Frederiksen, H., Skakkebaek, N. E., Wulf, H. C., and Andersson, A. M. (2008). Urinary excretion of phthalates and paraben after repeated whole-body topical application in humans. *International Journal of Andrology*, 31:118–30.

Jensen, L. J., Kuhn, M., Stark, M., Chaffron, S., Creevey, C., Muller, J., Doerks, T., Julien, P., Roth, A., Simonovic, M., *et al.* (2008). STRING 8—a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Research*, 37(suppl_1):D412–D416.

Jiang, J. and Zhai, C. (2007). An empirical study of tokenization strategies for biomedical information retrieval. *Information Retrieval*, 10:341–363.

Jin, Y. (2007). *Extracting cancer genomics knowledge from biomedical literature*. University of Pennsylvania.

Joachims, T. (2002). *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers.

Johnson, H. L., Baumgartner, W. A., Krallinger, M., Cohen, K. B., and Hunter, L. (2007). Corpus refactoring: a feasibility study. *Journal of Biomedical Discovery and Collaboration*, 2(1):4.

Jurafsky, D. (2000). *Speech & language processing*. Pearson Education India.

Kanehisa, M., Araki, M., Goto, S., Hattori, M., Hirakawa, M., Itoh, M., Katayama, T., Kawashima, S., Okuda, S., Tokimatsu, T., *et al.* (2008). KEGG for linking genomes to life and the environment. *Nucleic Acids Research*, 36(suppl 1):D480–D484.

KEMI (2015). Phthalates which are toxic for reproduction and endocrine-disrupting—proposals for a phase-out in Sweden. Report 4/15. Stockholm. Available: http://www.kemi.se/en.

Kiela, D., Guo, Y., Stenius, U., and Korhonen, A. (2015). Unsupervised discovery of information structure in biomedical documents. *Bioinformatics*, 31(7):1084–1092.

Kim, J.-D., Ohta, T., Pyysalo, S., Kano, Y., and Tsujii, J. (2009). Overview of BioNLP2009 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 Tsujii, J. (2003). GENIA corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl_1):i180–i182.

Kim, J.-D., Ohta, T., Tsuruoka, Y., Tateisi, Y., and Collier, N. (2004). Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 70–75. Association for Computational Linguistics.

Kim, J.-D., Wang, Y., Takagi, T., and Yonezawa, A. (2011). Overview of GENIA event task in BioNLP shared task 2011. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 7–15. Association for Computational Linguistics.

Kim, J.-D., Wang, Y., and Yasunori, Y. (2013). The GENIA event extraction shared task, 2013 edition-overview. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 8–15. Association for Computational Linguistics.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koonin, E. V. (2005). Orthologs, paralogs, and evolutionary genomics. *Annual Review of Genetics*, 39:309–338.

Koopman, B., Zuccon, G., Nguyen, A., Bergheim, A., and Grayson, N. (2015). Automatic ICD-10 classification of cancers from free-text death certificates. *International Journal of Medical Informatics*, 84(11):956–965.

Korhonen, A., Guo, Y., Baker, S., Yetisgen-Yildiz, M., Stenius, U., Narita, M., and Liò, P. (2015). *Improving Literature-Based Discovery with Advanced Text Mining*, pages 89–98. Springer International Publishing, Cham.

Korhonen, A., Ó'Séaghdha, D., Silins, I., Sun, L., Högberg, J., and Stenius, U. (2012). Text mining for literature review and knowledge discovery in cancer risk assessment and research. *PloS One*, 7(4):e33427.

Korhonen, A., Silins, I., Sun, L., and Stenius, U. (2009). The first step in the development of text mining technology for cancer risk assessment: identifying and organizing scientific evidence in risk assessment literature. *BMC Bioinformatics*, 10(1):303.

Kostoff, R. (2008). Where is the discovery in literature-based discovery? *Literature-Based Discovery*, pages 57–72.

Kostoff, R. N. and Briggs, M. B. (2008). Literature-related discovery (LRD): potential treatments for parkinson's disease. *Technological Forecasting and Social Change*, 75(2):226–238.

Krallinger, M., Erhardt, R. A.-A., and Valencia, A. (2005). Text-mining approaches in molecular biology and biomedicine. *Drug Discovery Today*, 10(6):439–445.

Krauthammer, M. and Nenadic, G. (2004). Term identification in the biomedical literature. *Journal of Biomedical Informatics*, 37(6):512–526.

Kulick, S., Bies, A., Liberman, M., Mandel, M., McDonald, R., Palmer, M., Schein, A., Ungar, L., Winters, S., and White, P. (2004). Integrated annotation for biomedical information extraction. In *HLT-NAACL 2004 Workshop: Linking Biological Literature, Ontologies and Databases*.

Kurata, G., Xiang, B., and Zhou, B. (2016). Improved neural network-based multi-label classification with better initialization leveraging label co-occurrence. In *Proceedings of NAACL-HLT*, pages 521–526.

Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. Q. (2015). From word embeddings to document distances. In Blei, D. and Bach, F., editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 957–966. JMLR Workshop and Conference Proceedings.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Increase.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174.

Larsson, K., Baker, S., Silins, I., Guo, Y., Stenius, U., Korhonen, A., and Berglund, M. (2017). Text mining for improved exposure assessment. *PloS One*, 12(3):e0173132.

Lazebnik, Y. (2010). What are the hallmarks of cancer? *Nature Reviews Cancer*, 10(4):232–233.

Le, Q. V. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196.

Leaman, R., Gonzalez, G., *et al.* (2008). BANNER: an executable survey of advances in biomedical named entity recognition. In *Pacific symposium on Biocomputing*, volume 13, pages 652–663. Big Island, Hawaii.

Leaman, R. and Lu, Z. (2016). Taggerone: joint named entity recognition and normalization with semi-markov models. *Bioinformatics*, 32(18):2839–2846.

LeCun, Y. and Bengio, Y. (1995). Convolutional networks for images, speech, and time series. *The Handbook of Brain Theory and Neural Networks*, 3361(10):1995.

Lee, C.-H., Chiu, H.-C., and Yang, H.-C. (2007). A platform of biomedical literature mining for categorization of cancer related abstracts. In *Second International Conference on Innovative Computing, Information and Control, 2007. ICICIC'07.*, pages 174–174. IEEE.

Lee, H.-J. (2014). OncoSearch: cancer gene search engine with literature evidence. *Nucleic Acids Research*.

Lei, C. and Ruan, J. (2012). A novel link prediction algorithm for reconstructing protein–protein interaction networks by topological similarity. *Bioinformatics*, 29(3):355–364.

Leitner, F., Mardis, S. A., Krallinger, M., Cesareni, G., Hirschman, L. A., and Valencia, A. (2010). An overview of BioCreative II. 5. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7(3):385–399.

Leser, U. and Hakenberg, J. (2005). What makes a gene name? named entity recognition in the biomedical literature. *Briefings in Bioinformatics*, 6(4):357–369.

Lewis, D. D., Yang, Y., Rose, T. G., and Li, F. (2004). RCV1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5(Apr):361–397.

Li, F., Zhang, M., Fu, G., and Ji, D. (2017). A neural joint model for entity and relation extraction from biomedical text. *BMC Bioinformatics*, 18(1):198.

Li, J., Zhang, Z., Li, X., and Chen, H. (2008). Kernel-based learning for biomedical relation extraction. *Journal of the Association for Information Science and Technology*, 59(5):756–769.

Limsopatham, N. and Collier, N. (2016). Modelling the combination of generic and target domain embeddings in a convolutional neural network for sentence classification. In *Proceedings of BioNLP 2016*, page 136.

Lin, J., Kuang, Q., Li, Y., Zhang, Y., Sun, J., Ding, Z., and Li, M. (2013). Prediction of adverse drug reactions by a network based external link prediction method. *Analytical Methods*, 5(21):6120–6127.

Lippincott, T., Ó'Séaghdha, D., and Korhonen, A. (2011). Exploring subdomain variation in biomedical language. *BMC Bioinformatics*, 12(1):212.

Lipscomb, C. E. (2000). Medical subject headings (MeSH). *Bulletin of the Medical Library Association*, 88(3):265.

Liu, H., Christiansen, T., Baumgartner Jr, W. A., and Verspoor, K. (2012). BioLemmatizer: a lemmatization tool for morphological processing of biomedical text. *Journal of Biomedical Semantics*, 3:3.

Liu, H. and Rastegar-Mojarad, M. (2016). Literature-based knowledge discovery. *Big Data Analysis for Bioinformatics and Biomedical Discoveries*, pages 233–248.

Liu, K., Hogan, W. R., and Crowley, R. S. (2011). Natural language processing methods and systems for biomedical ontology learning. *Journal of Biomedical Informatics*, 44(1):163–179.

Liu, Y., Liu, Z., Chua, T.-S., and Sun, M. (2015). Topical word embeddings. In *AAAI*, pages 2418–2424.

Lowe, H. J. and Barnett, G. O. (1994). Understanding and using the medical subject headings (MeSH) vocabulary to perform literature searches. *Journal of the American Medical Society*, 271(14):1103–1108.

Lu, Y., Liu, P., Wen, W., Grubbs, C. J., Townsend, R. R., Malone, J. P., Lubet, R. A., and You, M. (2011). Cross-species comparison of orthologous gene expression in human bladder cancer and carcinogen-induced rodent models. *American Journal of Translational Research*, 3(1):8.

Maglott, D., Ostell, J., Pruitt, K. D., and Tatusova, T. (2005). Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Research*, 33(suppl_1):D54–D58.

Manning, C. D., Schütze, H., *et al.* (1999). *Foundations of statistical natural language processing*. MIT Press.

McClosky, D. and Charniak, E. (2008). Self-training for biomedical parsing. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 101–104. Association for Computational Linguistics.

McCowan, I. A., Moore, D. C., Nguyen, A. N., Bowman, R. V., Clarke, B. E., Duhig, E. E., and Fry, M.-J. (2007). Collection of cancer stage data by classifying free-text medical reports. *Journal of the American Medical Informatics Association*, 14(6):736–745.

McDonald, J. H. (2009). *Handbook of biological statistics*, volume 2. Sparky House Publishing Baltimore, MD.

McGuire, S. (2016). World cancer report 2014. Geneva, Switzerland: World Health Organization, international agency for research on cancer, WHO Press, 2015. *Advances in Nutrition: An International Review Journal*, 7(2):418–419.

McInnes, B. T. (2004). *Extending the Log Likelihood Measure to Improve Collection Identification*. PhD thesis, University of Minnesota, Duluth.

Menon, A. K. and Elkan, C. (2011). Link prediction via matrix factorization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 437–452. Springer.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. In *Proceedings of ICLR*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111–3119.

Nam, J., Kim, J., Mencía, E. L., Gurevych, I., and Fürnkranz, J. (2014). Large-scale multi-label text classification—revisiting neural networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 437–452. Springer.

Nassif, H., Woods, R., Burnside, E., Ayvaci, M., Shavlik, J., and Page, D. (2009). Information extraction for clinical data mining: a mammography case study. In *IEEE International Conference on Data Mining Workshops, 2009. ICDMW'09.*, pages 37–42. IEEE.

Natale, D. A., Arighi, C. N., Barker, W. C., Blake, J. A., Bult, C. J., Caudy, M., Drabkin, H. J., D'Eustachio, P., Evsikov, A. V., Huang, H., *et al.* (2010). The Protein Ontology: a structured representation of protein forms and complexes. *Nucleic Acids Research*, 39(suppl_1):D539–D545.

Natarajan, J., Berrar, D., Dubitzky, W., Hack, C., Zhang, Y., DeSesa, C., Van Brocklyn, J. R., and Bremer, E. G. (2006). Text mining of full-text journal articles combined with gene expression analysis reveals a relationship between sphingosine-1-phosphate and invasiveness of a glioblastoma cell line. *BMC Bioinformatics*, 7(1):373.

Natarajan, S., Bangera, V., Khot, T., Picado, J., Wazalwar, A., Costa, V. S., Page, D., and Caldwell, M. (2017). Markov logic networks for adverse drug event extraction from text. *Knowledge and Information Systems*, 51(2):435–457.

Newhook, R. and Dormer, W. (1997). Environmental health criteria 195: Hexachlorobenzene.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y. (2011). Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 689–696.

Nguyen, A. N., Lawley, M. J., Hansen, D. P., Bowman, R. V., Clarke, B. E., Duhig, E. E., and Colquist, S. (2010). Symbolic rule-based classification of lung cancer stages from free-text pathology reports. *Journal of the American Medical Informatics Association*, 17(4):440–445.

Nguyen, D. X., Bos, P. D., and Massagué, J. (2009). Metastasis: from dissemination to organ-specific colonization. *Nature Reviews Cancer*, 9(4):274.

Nickel, M., Murphy, K., Tresp, V., and Gabrilovich, E. (2016). A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33.

Normanno, N., De Luca, A., Bianco, C., Strizzi, L., Mancino, M., Maiello, M. R., Carotenuto, A., De Feo, G., Caponigro, F., and Salomon, D. S. (2006). Epidermal growth factor receptor (EGFR) signaling in cancer. *Gene*, 366(1):2–16.

NRC (1983). *Risk assessment in the federal government: managing the process*. National Academies Press.

Ohta, T., Pyysalo, S., Rak, R., Rowley, A., Chun, H.-W., Jung, S.-J., Jeong, C.-h., Choi, S.-p., and Ananiadou, S. (2013). Overview of the pathway curation (PC) task of BioNLP shared task 2013. *BMC Bioinformatics*.

Okazaki, N. and Ananiadou, S. (2006). Building an abbreviation dictionary using a term recognition approach. *Bioinformatics*, 22(24):3089–3095.

Ongenaert, M., Van Neste, L., De Meyer, T., Menschaert, G., Bekaert, S., and Van Criekinge, W. (2007). PubMeth: a cancer methylation database combining text-mining and expert annotation. *Nucleic Acids Research*, 36(suppl_1):D842–D846.

Payne, P. R., Mendonça, E. A., Johnson, S. B., and Starren, J. B. (2007). Conceptual knowledge acquisition in biomedicine: A methodological review. *Journal of Biomedical Informatics*, 40(5):582–602.

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

Peng, Y., Bonifield, G., and Smalheiser, N. R. (2017). Gaps within the biomedical literature: Initial characterization and assessment of strategies for discovery. *Frontiers in Research Metrics and Analytics*, 2:3.

Petric, I., Ligeti, B., Gyorffy, B., and Pongor, S. (2014). Biomedical hypothesis generation by text mining and gene prioritization. *Protein and Peptide Letters*, 21(8):847–857.

Pihan, G. A. and Doxsey, S. J. (1999). The mitotic machinery as a source of genetic instability in cancer. In *Seminars in Cancer Biology*, volume 9, pages 289–302. Elsevier.

Poon, H., Toutanova, K., and Quirk, C. (2014). Distant supervision for cancer pathway extraction from text. In *Pacific Symposium on Biocomputing Co-Chairs*, pages 120–131.

Preiss, J. and Stevenson, M. (2016). The effect of word sense disambiguation accuracy on literature based discovery. *BMC Medical Informatics and Decision Making*, 16(1):57.

Preiss, J., Stevenson, M., and Gaizauskas, R. (2015). Exploring relation types for literature-based discovery. *Journal of the American Medical Informatics Association*, 22(5):987–992.

Preiss, J., Stevenson, M., and McClure, M. H. (2012). Towards semantic literature based discovery. In *2012 AAAI Fall Symposium Series: Information Retrieval and Knowledge Discovery in Biomedical Text*, volume 30, pages 7–18.

Pyysalo, S., Baker, S., Ali, I., Haselwimmer, S., Shah, T., Young, A., Högberg, J., Stenius, U., Narita, M., and Korhonen, A. (2017). LION LBD: a literature-based discovery system for cancer biology. Under review.

Pyysalo, S., Ginter, F., Heimonen, J., Björne, J., Boberg, J., Järvinen, J., and Salakoski, T. (2007). BioInfer: a corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(1):50.

Pyysalo, S., Ginter, F., Moen, H., Salakoski, T., and Ananiadou, S. (2013a). Distributional semantics resources for biomedical text processing. *Proceedings of LBM*.

Pyysalo, S., Ohta, T., and Ananiadou, S. (2013b). Overview of the cancer genetics (CG) task of BioNLP Shared Task 2013. In *BioNLP Shared Task 2013 Workshop*.

Quan, C., Wang, M., and Ren, F. (2014). An unsupervised text mining method for relation extraction from biomedical literature. *PLoS One*.

Rao, S., Marcu, D., Knight, K., and Hal, D. (2017). Biomedical event extraction using abstract meaning representation. *BioNLP 2017*, page 126.

Renner, R. (1997). European bans on surfactant trigger transatlantic debate. *Environmental Science & Technology*, 31(7):316A–320A.

Rimell, L. and Clark, S. (2009). Porting a lexicalized-grammar parser to the biomedical domain. *Journal of Biomedical Informatics*, 42(5):852–865.

Rinaldi, F., Ellendorff, T. R., Madan, S., Clematide, S., Van der Lek, A., Mevissen, T., and Fluck, J. (2016). BioCreative V track 4: a shared task for the extraction of causal network information using the biological expression language. *Database*, 2016.

Rinaldi, F., Kappeler, T., Kaljurand, K., Schneider, G., Klenner, M., Clematide, S., Hess, M., Von Allmen, J.-M., Parisot, P., Romacker, M., *et al.* (2008). Ontogene in BioCreative II. *Genome Biology*, 9(2):S13.

Roberts, R. J. (2001). PubMed Central: The GenBank of the published literature.

Rubner, Y., Tomasi, C., and Guibas, L. J. (2000). The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121.

Sam, H. M. and McInnes, B. (2017). Literature based discovery: models, methods, and trends. *Journal of Biomedical Informatics*.

Sang, S., Yang, Z., Li, Z., and Lin, H. (2015). Supervised learning based hypothesis generation from biomedical literature. *BioMed Research International*, 2015.

Schiff, P. and Horwitz, S. B. (1980). Taxol stabilizes microtubules in mouse fibroblast cells. *Proceedings of the National Academy of Sciences*, 77(3):1561–1565.

Sebastian, Y., Siew, E.-G., and Orimaye, S. O. (2017). Emerging approaches in literature-based discovery: techniques and performance review. *The Knowledge Engineering Review*, 32.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys (CSUR)*, 34(1):1–47.

Segura-Bedmar, I., Martinez, P., and de Pablo-Sánchez, C. (2010). Extracting drug-drug interactions from biomedical texts. *BMC Bioinformatics*, 11(S5):P9.

Settles, B. (2004). Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 104–107. Association for Computational Linguistics.

Settles, B. (2005). ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text. *Bioinformatics*, 21(14):3191–3192.

Severyn, A. and Moschitti, A. (2015). UNITN: Training deep convolutional neural network for twitter sentiment classification. In *Proceedings of SemEval 2015*, pages 464–469.

Sharma, V. K., Anquandah, G. A., Yngard, R. A., Kim, H., Fekete, J., Bouzek, K., Ray, A. K., and Golovko, D. (2009). Nonylphenol, octylphenol, and bisphenol-a in the aquatic environment: a review on occurrence, fate, and treatment. *Journal of Environmental Science and Health Part A*, 44(5):423–442.

Smalheiser, N. R., Torvik, V. I., and Zhou, W. (2009). Arrowsmith two-node search interface: A tutorial on finding meaningful links between two disparate sets of articles in MEDLINE. *Computer Methods and Programs in Biomedicine*, 94(2):190–197.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., *et al.* (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology*, 25(11):1251.

Sokolov, A. and Ben-Hur, A. (2010). Hierarchical classification of gene ontology terms using the gostruct method. *Journal of Bioinformatics and Computational Biology*, 8(02):357–376.

Sokolov, A., Funk, C., Graim, K., Verspoor, K., and Ben-Hur, A. (2013). Combining heterogeneous data sources for accurate functional annotation of proteins. *BMC Bioinformatics*, 14(3):S10.

Spasić, I., Livsey, J., Keane, J. A., and Nenadić, G. (2014). Text mining of cancer-related information: review of current status and future directions. *International Journal of Medical Informatics*, 83(9):605–623.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Statnikov, A., Wang, L., and Aliferis, C. F. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC Bioinformatics*, 9(1):319.

Stearns, M. Q., Price, C., Spackman, K. A., and Wang, A. Y. (2001). SNOMED clinical terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association.

Stenetorp, P., Soyer, H., Pyysalo, S., Ananiadou, S., and Chikayama, T. (2012). Size (and domain) matters: Evaluating semantic word space representations for biomedical text. In *Proceedings of SMBM 2012*.

Strzalkowski, T. (1999). *Natural language information retrieval*. MIT Press.

Sun, A. and Lim, E.-P. (2001). Hierarchical text classification and evaluation. In *Proceedings of the IEEE International Conference on Data Mining, 2001*, pages 521–528. IEEE.

Sun, L. and Korhonen, A. (2009). Improving verb clustering with automatically acquired selectional preferences. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2*, pages 638–647. Association for Computational Linguistics.

Swanson, D. R. (1987). Two medical literatures that are logically but not bibliographically connected. *Journal of the American Society for Information Science*, 38(4):228.

Swanson, D. R. (1988). Migraine and magnesium: eleven neglected connections. *Perspectives in Biology and Medicine*, 31(4):526–557.

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.

Szarvas, G., Vincze, V., Farkas, R., and Csirik, J. (2008). The bioscope corpus: annotation for negation, uncertainty and their scope in biomedical texts. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, pages 38–45. Association for Computational Linguistics.

Tanabe, L., Xie, N., Thom, L. H., Matten, W., and Wilbur, W. J. (2005). GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC Bioinformatics*, 6(1):S3.

Tang, B., Cao, H., Wang, X., Chen, Q., and Xu, H. (2014). Evaluating word representation features in biomedical named entity recognition tasks. *BioMed Research International*, 2014.

Teufel, S. and Moens, M. (2002). Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational Linguistics*, 28(4):409–445.

Thompson, P., McNaught, J., Montemagni, S., Calzolari, N., Del Gratta, R., Lee, V., Marchi, S., Monachini, M., Pezik, P., Quochi, V., *et al.* (2011a). The BioLexicon: a large-scale terminological resource for biomedical text mining. *BMC Bioinformatics*, 12(1):397.

Thompson, P., Nawaz, R., McNaught, J., and Ananiadou, S. (2011b). Enriching a biomedical event corpus with meta-knowledge annotation. *BMC Bioinformatics*, 12(1):393.

Tomasetti, C. and Vogelstein, B. (2015). Variation in cancer risk among tissues can be explained by the number of stem cell divisions. *Science*, 347:78–81.

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 Tsujii, J. (2005). Developing a robust part-of-speech tagger for biomedical text. In *Proceedings of the 10th Panhellenic Conference on Advances in Informatics*, PCI'05, pages 382–392, Berlin, Heidelberg. Springer-Verlag.

Tsuruoka, Y. and Tsujii, J. (2005). Bidirectional inference with the easiest-first strategy for tagging sequence data. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 467–474, Stroudsburg, PA, USA. Association for Computational Linguistics.

Vapnik, V. (1963). Pattern recognition using generalized portrait method. *Automation and Remote Control*, 24:774–780.

Varma, S. and Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, 7(1):91.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164.

Wan, X. (2007). A novel document similarity measure based on earth mover's distance. *Information Sciences*, 177(18):3718–3730.

Wang, D. and Lippard, S. J. (2005). Cellular processing of platinum anticancer drugs. *Nature Reviews Drug Discovery*, 4(4):307.

Wei, C.-H., Peng, Y., Leaman, R., Davis, A. P., Mattingly, C. J., Li, J., Wiegers, T. C., and Lu, Z. (2016). Assessing the state of the art in biomedical relation extraction: overview of the BioCreative V chemical-disease relation (CDR) task. *Database*, 2016.

Wilhelm, S., Carter, C., Lynch, M., Lowinger, T., Dumas, J., Smith, R. A., Schwartz, B., Simantov, R., and Kelley, S. (2006). Discovery and development of Sorafenib: a multikinase inhibitor for treating cancer. *Nature Reviews Drug Discovery*, 5(10):835–844.

Wittassek, M. and Angerer, J. (2008). Phthalates: metabolism and exposure. *International Journal of Andrology*, 31:131–8.

Wittassek, M., Koch, H. M., Angerer, J., and Bruning, T. (2011). Assessing exposure to phthalates—the human biomonitoring approach. *Molecular Nutrition & Food Research*, 55:7–31.

World Health Organization (2009). International statistical classification of diseases and related health problems.

Wormuth, M., Scheringer, M., Vollenweider, M., and Hungerbuhler, K. (2006). What are the sources of exposure to eight frequently used phthalic acid esters in Europeans? *Risk Analysis*, 26:803–24.

Xie, B., Ding, Q., Han, H., and Wu, D. (2013). miRCancer: a microRNA—cancer association database constructed by text mining on literature. *Bioinformatics*, 29(5):638–644.

Xu, R. and Wang, Q. (2013). Large-scale extraction of accurate drug-disease treatment pairs from biomedical literature for drug repurposing. *BMC Bioinformatics*, 14(1):181.

Xu, W., Sun, H., Deng, C., and Tan, Y. (2017). Variational autoencoder for semi-supervised text classification. In *AAAI*, pages 3358–3364.

Yan, Y., Yin, X.-C., Li, S., Yang, M., and Hao, H.-W. (2015). Learning document semantic representation with hybrid deep belief network. *Computational Intelligence and Neuroscience*, 2015.

Yang, H.-T., Ju, J.-H., Wong, Y.-T., Shmulevich, I., and Chiang, J.-H. (2017). Literature-based discovery of new candidates for drug repurposing. *Briefings in Bioinformatics*, 18(3):488–497.

Yang, Y. and Chute, C. (1992). An application of least squares fit mapping to clinical classification. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, page 460. American Medical Informatics Association.

Yetisgen-Yildiz, M. and Pratt, W. (2006). Using statistical and knowledge-based approaches for literature-based discovery. *Journal of Biomedical Informatics*, 39(6):600–611.

Yogatama, D., Faruqui, M., Dyer, C., and Smith, N. (2015). Learning word representations with hierarchical sparse coding. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 87–96.

Yogatama, D. and Smith, N. A. (2014). Linguistic structured sparsity in text categorization. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Young, L., Listgarten, J., Trotter, M., Andrew, S., and Tron, V. (2008). Evidence that dysregulated DNA mismatch repair characterizes human nonmelanoma skin cancer. *British Journal of Dermatology*, 158(1):59–69.

Zhang, M.-L. and Zhou, Z.-H. (2006). Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1338–1351.

Zhang, R., Cairelli, M. J., Fiszman, M., Kilicoglu, H., Rindflesch, T. C., Pakhomov, S. V., and Melton, G. B. (2014). Exploiting literature-derived knowledge and semantics to identify potential prostate cancer drugs. *Cancer Informatics*, 13(Suppl 1):103.

Zhang, S. and Elhadad, N. (2013). Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. *Journal of Biomedical Informatics*, 46(6):1088–1098.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, pages 649–657.

Zhang, Y. and Liu, B. (2007). Semantic text classification of emergent disease reports. *Knowledge Discovery in Databases: PKDD 2007*, pages 629–637.

Zhang, Y. and Wallace, B. (2015). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*.

Zhao, Y. and Adjei, A. A. (2015). Targeting angiogenesis in cancer therapy: moving beyond vascular endothelial growth factor. *The Oncologist*, 20(6):660–673.

Zhou, D., Zhong, D., and He, Y. (2014). Biomedical relation extraction: from binary to complex. *Computational and Mathematical Methods in Medicine*, 2014.

Zhu, F., Patumcharoenpol, P., Zhang, C., Yang, Y., Chan, J., Meechai, A., Vongsangnak, W., and Shen, B. (2013). Biomedical text mining and its applications in cancer research. *Journal of Biomedical Informatics*, 46(2):200–211.

Zweigenbaum, P., Demner-Fushman, D., Yu, H., and Cohen, K. B. (2007). Frontiers of biomedical text mining: current progress. *Briefings in Bioinformatics*, 8(5):358–375.

Appendix A

Supplementaries

Table A.1: Description of the Hallmarks of Cancer (HoC) taxonomy.

Hallmark	Description
Activating Invasion and metastasis	Cancer cells can break away from their site of origin to invade surrounding tissue and spread to distant body parts.
Invasion	The infiltration or penetration of neoplasms to actively destroy the surrounding tissues of its primary site.
Metastasis	The transfer of a neoplasm from one organ or part of the body to another remote from the primary site.
Avoiding immune destruction	Cancer cells are invisible to the immune system.
Immune response	Any immune system process that recognises and defends an organism with a calibrated response against any potential internal or foreign threat.
Immunosuppression	Any process that stops, prevents, or reduces the frequency, rate, or extent of the immune response, the immunological reaction of an organism to an immunogenic stimulus.
Deregulating cellular energetics	Most cancer cells use abnormal metabolic pathways to generate energy. For example, they may generate energy *via* glucose fermentation even when enough energy is present to properly respire.
Glycolysis / Warburg effect	The chemical reactions and pathways resulting in the breakdown of a carbohydrate into pyruvate, with the concomitant production of a small amount of ATP and the reduction of NADP to NADPH.
Enabling replicative immortality	Noncancer cells die after a certain number of divisions. Cancer cells, however, are capable of indefinite growth and division (immortality).
Immortalization	A phenomenon under which the modifications in a cell enable it to reproduce indefinitely.
Senescence	A biological aging process that involves gradual dysfunctional changes by which a cell become less capable of maintaining physiological functions (*e.g.* normal cells lose the ability to divide through irreversible cell cycle arrest).

Hallmark	Description
Evading growth suppressors	Cells have processes that halt growth and division. In cancer cells, these processes are altered so that they don't effectively prevent all division.
Deregulating cell cycle	Deregulating the progression of biochemical and morphological phases and events that occur in a cell during successive cell replication or nuclear replication events.
Evading contact inhibition	A growth control mechanism by which cells stop growing or dividing depending on the conditions (*e.g.* increased cell density).
Genomic instability and mutation	Cancer cells generally have severe chromosomal abnormalities, which worsen as the disease progresses.
DNA damage	Drug-, chemical-, radiation-induced, or spontaneous injuries to the chemical structure of DNA (*e.g.* break in the strand of DNA).
DNA adducts	DNA adducts are formed after covalent bonding between a molecule and the DNA macromolecule that may interfere with DNA replication and repair.
DNA strand breaks	A DNA strand break involves one or more disruptions of the covalent linkages among phosphor-deoxyribose moieties within the sugar–phosphate backbone in one or in both strands of a DNA molecule.
DNA repair	A collection of cellular processes that maintain the original DNA sequence by identifying and correcting the damage to the DNA molecule—for example, base excision repair, double strand break repair, and mismatch repair.
Mutation	Any permanent change in the DNA sequence of a cell or the genome of an organism, which can result from radiation, viral infection, transposition, mutagenic chemicals, and/or from errors during DNA replication or meiosis.
Inducing angiogenesis	Cancer cells are able to initiate angiogenesis, the process by which new blood vessels are formed, thus ensuring the supply of oxygen and other nutrients.
Deregulating angiogenesis	A physiological process of blood vessel formation through which new vessels emerge from the proliferation of pre-existing blood vessels.
Angiogenic factors	Multiple factors including soluble growth and differentiative factors, extracellular matrix components, membrane-bound receptors, and intracellular signalling molecules which contribute to angiogenesis.
Resisting cell death	Apoptosis is a mechanism by which cells are programmed to die in the event that they become damaged. Cancer cells are able to bypass these mechanisms.
Apoptosis	A highly regulated and controlled cell death process which begins when a cell receives an internal (*e.g.* DNA damage) or external signal (*e.g.* an extracellular death ligand), and proceeds through a series of biochemical events (signalling pathway phase) which trigger an execution phase.

Hallmark	Description
Autophagy	The natural process in which cells digest or disassemble unnecessary or dysfunctional cellular components, thereby allowing for recycling of macromolecular constituents under conditions of cellular stress and remodelling the intracellular structure for cell differentiation.
Necrosis	A premature cell death that involves unregulated digestion of cell component as a result of infection, toxins, or trauma. It is morphologically characterised by an increasingly translucent cytoplasm, swelling of organelles, minor ultrastructural modifications of the nucleus, and increased cell volume (oncosis), culminating in the disruption of the plasma membrane and subsequent loss of intracellular contents.
Sustaining proliferative signaling	Healthy cells require molecules that act as signals for them to grow and divide. Cancer cells, on the other hand, are able to grow without these external signals.
Cell cycle	The progression of biochemical and morphological phases and events that occur in a cell during successive cell replication or nuclear replication events.
Growth signals	Any process that results in a change in state or activity of a cell (in terms of movement, secretion, enzyme production, gene expression, *etc.*) as a result of a growth factor stimulus.
Growth signals – downstream signalling	The temporal and mechanistic order of cellular and molecular events. For example, in signal transduction, the second messenger acts downstream to activation of cell membrane receptors.
Receptors	The growth factor receptors (such as EGFR) are cell membrane receptors that after being activated (by, for example, a hormone or a growth factor) initiate the signal transduction pathway.
Tumor promoting inflammation	Inflammation affects the microenvironment surrounding tumours, contributing to the proliferation, survival, and metastasis of cancer cells.
Inflammation	The immediate defensive reaction of body tissues to infection or injury caused by chemical or physical agents.
Oxidative stress	A condition characterised by an imbalance between the systemic manifestation of reactive oxygen species and the ability of a biological system to readily detoxify the reactive intermediates or repair the resulting damage to cellular environment.

Table A.2: The number of features for each class in the Hallmarks of Cancer taxonomy after feature selection.

Hallmark	LBOW	GR	NER	VC	NB	MeSH	Chem	Total
Activating invasion and metastasis	1052	253	161	114	220	172	43	2015
Invasion	682	128	109	105	115	100	25	1264
Metastasis	732	125	82	106	139	98	21	1303
Avoiding immune destruction	498	59	55	99	48	68	14	841
Immune response	373	38	42	88	32	50	8	631
Immunosuppression	217	17	17	79	21	25	3	379
Cellular energetics	410	53	31	103	54	68	20	739
Glycolysis / Warburg effect	386	49	31	99	51	59	19	694
Enabling replicative immortality	506	53	54	105	59	80	26	883
Immortalization	264	18	27	91	25	35	8	468
Senescence	328	36	41	94	37	53	17	606
Evading growth suppressors	863	189	108	119	176	156	55	1666
Deregulating cell cycle checkpoints	583	103	77	108	116	102	35	1124
Cell cycle	559	97	72	107	111	93	30	1069
Evading contact inhibition	399	43	37	103	58	45	7	692
Genomic instability and mutation	1215	235	100	126	188	216	77	2157
DNA damage	753	110	54	119	97	107	34	1274
Adducts	191	6	3	77	9	16	4	306
Strand breaks	327	26	20	99	35	28	6	541
DNA repair mechanisms	509	68	35	108	69	84	31	904
Mutation	463	40	30	108	51	65	15	772
Inducing angiogenesis	590	96	82	105	93	97	28	1091
Deregulating angiogenesis	579	82	82	104	89	91	24	1051
Angiogenic factors	355	39	52	88	47	51	14	646
Resisting cell death	1403	380	215	119	289	262	114	2782
Apoptosis	1155	312	179	113	260	217	94	2330
Autophagy	254	12	29	65	35	30	11	436
Necrosis	352	17	16	88	37	31	3	544
Sustaining proliferative signalling	1471	435	302	121	355	280	129	3093
Cell cycle	761	131	108	112	145	123	43	1423
Growth factors growth promoting signals	686	128	121	111	130	134	55	1365
Downstream signalling	402	57	57	91	78	75	29	789
Receptors	767	132	121	111	151	118	44	1444
Tumor promoting inflammation	843	152	99	111	122	121	34	1482
Immune response	221	16	14	79	17	25	3	375
Inflammation	786	138	92	111	110	110	30	1377
Oxidative stress	395	36	33	87	49	50	15	665
Average	604	106	75	102	100	96	32	1114

Table A.3: The number of features for each class in the Exposure taxonomy after feature selection.

Class	LBOW	GR	NE	VC	NB	MeSH	Chem	Total
Biomonitoring	4785	3544	352	128	1395	754	253	11211
Effect marker	4244	2965	312	127	1177	626	216	9667
Biomarker	361	66	23	67	38	56	19	630
Gene	2453	1300	141	122	521	355	111	5003
Molecule	1390	532	47	109	266	186	34	2564
- Lipid	668	197	43	91	85	86	35	1205
- Other molecule	612	106	14	92	66	82	21	993
- Protein	307	52	13	80	32	37	13	534
Other effect biomarker	2310	1164	124	120	482	314	120	4634
Oxidative stress	2889	1678	190	122	715	508	164	6266
Physiological parameter	1743	733	146	116	363	313	129	3543
Exposure biomarker	637	141	39	96	79	96	25	1113
Adipose tissue	1461	555	121	115	271	248	103	2874
Blood	447	84	19	88	54	69	30	791
Hair/nail	724	181	37	97	99	106	40	1284
Breast milk	1167	352	99	112	200	179	74	2183
Other tissue	357	44	16	73	35	38	11	574
Placenta	316	48	16	80	30	39	14	543
Urine	2100	1018	78	121	434	347	77	4175
Exposure Routes	4574	3356	248	130	1297	694	211	10510
Combined	715	156	18	98	91	98	31	1207
Dermal Exposure	773	142	15	99	94	79	29	1231
Inhalation	2607	1349	89	123	525	379	111	5183
Modeled exposure	1064	407	30	109	159	136	14	1919
Other inhalation	1308	369	35	107	191	178	58	2246
Personal air	974	221	25	99	124	104	32	1579
Oral intake	3216	2067	171	126	830	461	127	6998
Drinking water	1338	501	44	117	250	191	45	2486
Dust	1057	293	43	97	163	145	53	1851
Food	1932	997	87	117	411	270	78	3892
Products	803	193	18	97	117	99	29	1356
Soil	652	124	5	84	86	80	24	1055
Average	1328	624	67	100	271	193	61	2645

Table A.4: The number of features for each label used by CHAT classifiers after feature selection.

Hallmark	LBOW	Bigram	Trigram	VC	NE	MeSH	Chem	SD	Total
Activating invasion and metastasis	1922	718	114	115	212	172	43	1922	3296
Invasion	1209	365	50	103	141	100	25	1209	1993
Metastasis	1312	369	55	111	105	98	21	1312	2071
Avoiding immune destruction	900	162	18	102	73	68	14	900	1337
Immune response	690	103	9	89	57	50	8	690	1006
Immunosuppression	429	44	2	82	21	25	3	429	606
Cellular energetics	797	164	17	100	43	68	20	797	1209
Glycolysis / Warburg effect	744	153	15	97	39	59	19	744	1126
Enabling replicative immortality	931	172	11	105	67	80	26	931	1392
Immortalization	484	54	2	91	35	35	8	484	709
Senescence	647	109	5	94	48	53	17	647	973
Evading growth suppressors	1588	508	78	120	147	154	54	1588	2649
Deregulating cell cycle checkpoints	1106	304	50	106	101	101	34	1106	1802
Cell cycle	1047	289	44	105	96	91	28	1047	1700
Evading contact inhibition	765	116	13	104	49	45	7	765	1099
Genomic instability and mutation	2179	652	95	127	150	216	77	2179	3496
DNA damage	1414	320	40	118	76	107	33	1414	2108
Adducts	397	18	1	75	10	16	4	397	521
Strand breaks	633	87	5	93	32	27	5	633	882
DNA repair mechanisms	1008	174	22	105	51	84	31	1008	1475
Mutation	857	96	3	108	41	65	15	857	1185
Inducing angiogenesis	1104	266	46	104	105	97	28	1104	1750
Deregulating angiogenesis	1079	253	43	103	104	91	24	1079	1697
Angiogenic factors	669	119	20	88	66	51	14	669	1027
Resisting cell death	2551	1035	215	119	295	261	114	2551	4590
Apoptosis	2081	837	173	114	246	216	93	2081	3760
Autophagy	441	73	5	64	33	30	11	441	657
Necrosis	733	66	2	89	25	31	3	733	949
Sustaining proliferative signalling	2576	1167	239	122	390	280	129	2576	4903
Cell cycle	1359	423	68	112	144	123	43	1359	2272
Growth factors growth promoting signals	1270	357	51	111	159	134	55	1270	2137
Downstream signalling	740	176	19	91	82	75	29	740	1212
Receptors	1338	397	54	111	158	118	44	1338	2220
Tumor promoting inflammation	1808	520	64	115	157	147	40	1808	2851
Immune response	422	42	2	79	22	25	3	422	595
Inflammation	1730	488	58	113	146	141	39	1730	2715
Oxidative stress	1054	213	20	98	69	74	23	1054	1551
Average	1136	308	47	102	103	98	32	1136	1825

Appendix B

Software and resources

B.1 External software and resources

The following is a list of external resources (corpora, ontologies, software and tools) that have been used throughout this dissertation.

ABNER http://pages.cs.wisc.edu/bsettles/abner/

C&C http://groups.inf.ed.ac.uk/ccg/software.html

Chart.js http://chartjs.org/

Cytoscape.js http://js.cytoscape.org/

Flask http://flask.pocoo.org/

GENIA tagger http://www.nactem.ac.uk/GENIA/tagger/

Lucene http://lucene.apache.org/

PostgreSQL https://www.postgresql.org/

Protein Ontology http://pir.georgetown.edu/pro/

PubMed https://www.ncbi.nlm.nih.gov/pubmed/

PubTator http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/PubTator/

Semantic UI https://semantic-ui.com/

Scikit-learn http://scikit-learn.org/

UniProt http://www.uniprot.org/

word2vec https://code.google.com/archive/p/word2vec/

B.2 Developed software and resources

The following is a list of resources and software developed and publicly released during the work described in this dissertation.

1. The Hallmarks of Cancer (HoC) taxonomy corpus:
   https://www.cl.cam.ac.uk/~sb895/HoC.html
2. The entire 2016 `PubMed` release annotated with hallmarks using our CHAT classifier:
   http://chat.lionproject.net/static/pubmed2016_annotated_with_hallmarks.tar.gz
3. Software code for NLP pipeline described in Section 3.2.2:
   https://www.cl.cam.ac.uk/~sb895/HoCScripts.zip
4. The exposure taxonomy corpus and classier software:
   https://figshare.com/articles/Corpus_and_Software/4668229
5. cancer hallmark analytics tool (CHAT):
   http://chat.lionproject.net/
6. CHAT web application software code:
   https://github.com/cambridgeltl/chat
7. CHAT classifier software code:
   https://github.com/cambridgeltl/chat-classifier
8. HoC CNN (Section 4.2.2) classifier code:
   https://cambridgeltl.github.io/cancer-hallmark-cnn/
9. LION LBD tool:
   http://lbd.lionproject.net/