Distributed Representation of Biomedical Words for Drug Repositioning

Dissertation

Distributed Representation of Biomedical Words for Drug Repositioning

Graduate School of
Natural Science & Technology
Kanazawa University

Division of Electrical Engineering
and Computer Science

Student ID No.: 1323112003

Name: Ngo Duc Luu

Chief advisor: Professor Kenji Satou

Date of Submission: January 8th, 2016

Abstract

It can be said that the development of new and effective drugs becomes more essential for all pharmaceutical companies. However, the process of drug development usually requires many clear steps. Through the long time of research and development including clinical trial, the candidate chemical compounds are screened from thousands to one. It means that hundreds millions dollars' cost and over 15 years' time are needed. To make matters worse, success in the development is not guaranteed and so many failure projects exist. So drug development includes both high cost and high risk.

To avoid this problem, drug repositioning is actively studied. Briefly saying, drug repositioning is reuse of existing drugs for other purposes. Since all existing drugs have already been developed and sold in the market, it is possible to greatly reduce the cost and time for research and development (R&D). One famous example of drug repositioning is about thalidomide. At first, it was developed for good sleep. Due to the serious side effect for pregnant women, once it was withdrawn. However, it was repositioned as anti-cancer drug, and actually used for cancer treatment.

Besides biomedical experiments, computational methods are developed for drug repositioning. Most of them adopt network-based algorithms and combination of various databases including gene expression and pathway data. On the other hand, it is also suggested that text mining has much potential for drug repositioning.

In this study, a text mining approach to the discovery of unknown drug-disease relation was tested. Starting from over 3 million PubMed abstracts related to cancer, biomedical named entities (i.e. drugs, genes, proteins, etc.) were first recognized and the relations among them were extracted. Biomedical ontologies such as PharmGKB, MeSH, DrugBank, and CTD databases were utilized for the sources of semantic information. Using a word embedding algorithm, senses of over 1.7 million words were well represented in sufficiently short feature vectors. Through various analysis including clustering and classification, feasibility of our approach was tested. Finally, our trained classification model achieved 87.6% accuracy in the prediction of drug-disease relation in cancer treatment and succeeded in discovering novel drug-disease relations that actually reported in recent studies.

We strongly believe that word embedding is effective for representing sense of all words in large amount of cancer-related PubMed abstracts. Furthermore, concatenation of word vectors of drugs and diseases well represents their relations and could be used for finding candidate anti-cancer drugs for repositioning by classification.

Acknowledgments

In this time, I really want to send my deepest thanks to everyone, who has helped and encouraged me during the time I have studied and lived in this lovely university, Kanazawa University.

First of all, from the depth of my heart, I want to express my deep respect for my supervisor, Professor Kenji Satou for all his enormous help and exceptional guidance throughout the research. I am very proud and lucky since I have a wonderful advisor like him in my life.

I owe my sincere gratitude to all the committee members, Professor Kenji Satou, Professor Haruhiko Kimura, Professor Takeshi Fukuma, Associate Professor Yoichi Yamada, and Associate Professor Hidetaka Nambo for reading my thesis and giving the useful comments.

Special thanks to the Board of Mekong 1000 Project for donating the scholarship to me. It created a good chance for me to explore and enjoy the life in a beautiful and modern Japan.

I wish to thank Kanazawa University for giving me an opportunity to study here. In particular, I also want to offer my sincere thanks to all the staffs of Kanazawa University for their enthusiasm. I strongly believe that my studying must be very hard without their help.

Furthermore, I am also very grateful to my colleagues in Bac Lieu University, especially to Dr. Dao Hoang Nam for his great supports. He always encouraged and helped me to overcome all difficulties in my life to participate the PhD training program in oversea.

My gratitude goes to Vietnamese Student’s Union in Kanazawa City (Vietkindai) for supporting me a lot. Especially, I want to thank two sincere Japanese uncles, Abe-san and Tetsuo-san, for their help. For long time, I have considered Vietkindai as my second family.

I also am very happy to be a member of Bioinformatics Laboratory, Kanazawa University. I would like to send my appreciation to Lan Anh, Tu Kien, Vu Anh, Sergey, Ngoc Giang, Dau, Rosi, Bahriddin, Reza, and Japanese friends for the wonderful moments we had together.

Finally, I do not forget to send special thanks all members in my family, especially my darling, my daughter and my cute son who are always cheering up and encouraging me.

Thank you!

Contents

Abstract i

Acknowledgments... iii

Chapter 1 Introduction 1

1.1 Overview of dissertation 2

1.2 Objectives 3

1.3 Contributions 3

1.4 Thesis organization 4

Chapter 2 Related Works 5

2.1 Drug repositioning 6

2.2 Biomedical text mining 7

2.2.1 Basic concepts 7

2.2.2 The process of text mining 8

2.2.3 Approaches for text representation 11

2.2.4 Similarity measures 14

2.2.5 Text mining algorithms 15

Chapter 3 Materials and Methods 22

3.1 Overview of processing pipeline 23

3.2 Resources and tools 23

3.2.1 Biomedical databases 23

3.2.2 Tools 25

3.3 Data preparation 26

3.3.1 Cancer-related corpus from PubMed abstracts 26

3.3.2 Parsing sentences 27

3.3.3 Named entity recognition and conversion into single words 28

3.3.4 Word embedding 29

3.3.5 Combination of word vectors and database information...................... 30

3.3.6 Generation of the special vector for drug-disease relations 31
3.4 Algorithms 32
3.4.1 Checking the distribution of biomedical word vectors 32
3.4.2 Clustering for drugs and diseases 32
3.4.3 Classification for drug-disease relations 32
3.4.4 Cross-validation 32
3.4.5 Methods for the class-imbalance problem 33
Chapter 4 Experimental Results and Discussion 35
4.1 Distribution of drug-disease-gene vectors 36
4.2 Cluster analysis for drug and disease vectors 37
4.3 Classification of drug-disease relations 40
4.3.1 Applicability of simple word analogy to drug-disease relations 40
4.3.2 Classification of unseen drug-disease relations 41
Chapter 5 Conclusion and Future Works 43
5.1 Dissertation summary 44
5.2 Future works 45
Bibliography 46

List of Figures

Figure 2.1 Main strategies of drug repositioning. 7

Figure 2.2 Related fields of text mining. 8

Figure 2.3 CBOW model and Skip-gram model. 13

Figure 2.4 An example of word analogy. 14

Figure 2.5 Perceptron as the simplest neural network. 17

Figure 2.6 General architecture of the multi-layer neural network. 17

Figure 2.7 An example of dendrogram. 19

Figure 2.8 The process of the rule-based or dictionary-based approach. 20

Figure 3.1 Overview of processing pipeline. Box colors indicate: light blue for corpus, light green for databases, yellow for word vectors, and pink for concatenated word vectors. (a), (b), (c), and (d) corresponding to four phases of processing pipeline. 23

Figure 3.2 An example of splitting an abstract into sentences. 27

Figure 3.3 An example of POS tagging, base form conversion, and filtering by POS category. 27

Figure 3.4 Single word replacement for term “yolk sac tumor” in a sentence. 28

Figure 3.5 The concatenation of drug and disease word vectors to create drug-disease relation vector. If the relation is written in CTD with therapeutic evidence, it is labeled “TRUE” (otherwise, “FALSE”). 31

Figure 3.6 A strategy for solving the class imbalance problem. 34

Figure 4.1 Distribution of word vectors visualized through PCA and 3D plot. Top panel: blue, red, green colors indicate word vectors for drugs, diseases, and genes. Bottom panel: color gradation from light blue to light pink indicates the frequency of words (from rare to frequent). 36

Figure 4.2 Result of hierarchical clustering on drugs. Red, green, blue, yellow colors for characters indicate that the drugs are classified in ATC codes as “L01:Antineoplastic Agents”, “L02:Endocrine Therapy”, “L03:Immunostimulants”, and “L04:Immunosuppressants”, respectively. 38

Figure 4.3 Result of hierarchical clustering on diseases. Green, yellow, red, dodgerblue, pink, blue, and springgreen colors for characters indicate that the diseases are classified in MeSH tree numbers as “C04.182:Cysts”, “C04.445:Hamartoma”, “C04.557:Neoplasms by Histologic Type”, “C04.588:Neoplasms by Site”, “C04.697:Neoplastic Processes”, “C04.730:Paraneoplastic Syndromes”, and “C04.834:Precancerous Conditions”. 39

Figure 4.4 Distribution of displacement vectors for cancer-related drug-disease relations in CTD. Blue and red points represents anti-cancer drugs and cancer-related diseases. 40

List of Tables

Table 2.1 Text representation by Boolean model. 11
Table 2.2 Text representation by vector space model. 12
Table 4.1 Accuracy of classifying correct and incorrect drug-disease relations by SVM 41

Chapter 1

Introduction

In the first chapter we would like to introduce generally about our research which will be shown more detailed in the next chapters. Then, we briefly present the objectives of this study. Next, the main contributions of my dissertation are mentioned before we finish this chapter with the organization of the dissertation.

1.1 Overview of dissertation

For pharmaceutical companies, the development of new, effective, and highly-demanded drugs is important. To address this issue, hundreds million dollars and 10 or more years for research and development (R&D) and clinical trial are typically required. Structure-based drug design (SBDD) is actively studied to reduce the cost and time by *in-silico* screening of candidate chemicals [1, 2], however, still it requires long time for tests on animals and human. Beside the high cost and long time, the development is not always guarantied and so many failure projects exist. It means that pharmaceutical development includes both high cost and high risk.

Against such a background, a concept of drug repositioning (or drug repurposing, re-profiling, etc.) is attracting much interest and expectation from academic researchers and pharmaceutical companies [3]. One of the famous examples of drug repositioning is a treatment of multiple myeloma by thalidomide that was initially developed for relieving nausea and vomiting in pregnancy. Since drug repositioning means reuse of approved drugs for another purpose, their safety and method of production have already been confirmed.

In recent years, besides biomedical experiments, computational methods are developed for drug repositioning. Most of them adopt network-based algorithms and combination of various databases including gene expression and pathway data [4]. On the other hand, it is also suggested that text mining has much potential for drug repositioning. In biomedical text mining, named entities (genes, proteins, etc.) are recognized and the relations among them are extracted (e.g. "Gefitinib"<inhibit>"EGFR"). Additionally, biomedical ontologies or WordNet [5] are utilized for the sources of semantic information.

In this study, we applied word embedding, implemented as word2vec [6-8], for efficient representation of semantic information of words in a sufficiently large subset of PubMed abstracts. Through the clustering and classification experiments especially on anti-cancer drugs and cancer-related diseases, it is suggested that the word vectors, generated by word embedding for drugs and diseases, are representing rich semantic information and promising for drug repositioning.

1.2 Objectives

Since discovering new indications of existing drugs plays an important role for pharmaceutical and academic institutes, our research aims to apply available biomedical databases and text mining techniques to the extraction of new knowledge about anti-cancer drugs from large amount of PubMed abstracts.

The main objective of this thesis is to build a model which can predict unseen drug-disease relations based on biomedical word vectors and machine learning algorithms.

1.3 Contributions

In past decades, studies related to drug repositioning may contribute a lot of benefits for the development of new and effective drugs, especially for the development of highly-demanded drugs such as anti-cancer drugs or anti-HIV drugs.

This study mainly contributes to the following matters:

The first application of word embedding for drug repositioning

Unlike previous drug repositioning approaches, this study firstly applies word embedding model for the vector representation of biomedical words such as drugs, diseases, and genes. Word embedding, also called as distributed representation of words, is a new and effective word vector model which is broadly used in these years. Embedding words for drugs and diseases well reflects the meaning of them.

Understanding the distribution of biomedical word sense

From the result of cluster analysis for drug, disease, and gene word vectors, we can see the clear separation of word senses. This is essential for predicting the meaning and class labels of biomedical entities like drugs, diseases, and genes.

Generating new hypotheses about different indications of existing drugs

With trained classifiers, we can predict new drug-disease relations which are considered as new hypotheses related to the cancer treatment. From these hypotheses, along with pharmaceutical experts' support, new indications of existing drugs may be found to cure cancer or related diseases.

Reducing the cost and time to develop new anti-cancer drugs

In contrast of traditional drug repositioning methods, the uniqueness of this study is that it is purely based on text mining. Therefore, it can be said that this study significantly contributes to the reduction of time and money for drug development.

1.4 Thesis organization

This thesis is organized into five chapters, including the current chapter introducing the context, objectives, contributions, and organization of our research. The remaining chapters are organized as follows:

Chapter 2 reviews the backgrounds related to drug repositioning, text mining, and resources applied in the process of experiments.

Chapter 3 explains about the overview of processing pipeline, materials, and algorithms used for cluster analysis and classification.

Chapter 4 shows and discusses the obtained results from experiments.

Chapter 5 summarizes the dissertation by giving conclusion of achievement and presents about the future works.

Chapter 2

Related Works

This chapter aims to briefly present fundamental knowledge and works related to drug repositioning and biomedical text mining for drug development. Then, we introduce some biomedical resources and tools which have been used in our research.

2.1 Drug repositioning

For most pharmaceutical companies or institutions, finding the process for developing new and effective drugs is essential. However, the ratio of successfully identified drugs is quite low because the process of drug discovery often requires long time, high cost and many steps to bring new drugs to the market.

Through the long time of research and development including clinical trial, so many candidate chemical compounds are screened carefully in order to find lead compounds having good reactivity against pathways and single therapeutic targets. It means that hundreds millions dollars' cost and over 15 years' time are needed for this process. To make matters worse, success of the development is not always guaranteed and still so many failure projects exist. Therefore, we can say that drug development includes both high cost and high risk.

Against such a background, a concept of drug repositioning (or drug repurposing, reprofiling, etc.) is attracting much interest and expectation from academic researchers and pharmaceutical companies [3]. Briefly saying, drug repositioning is reuse of existing drugs for other purposes. One of the famous examples of drug repositioning is a treatment of multiple myeloma by thalidomide that was initially developed for relieving nausea and vomiting in pregnant women. Since drug repositioning means reuse of approved drugs for other purposes, their safety and method of production have already been confirmed.

Besides biomedical experiments, computational methods are developed for drug repositioning. Most of them adopt network-based algorithms and combination of various databases including gene expression and pathway data [4]. On the other hand, it is also suggested that text mining has much potential for drug repositioning as shown in figure 2.1. In biomedical text mining, named entities (genes, proteins, diseases, drugs, etc.) are recognized and the relations among them are extracted. Additionally, biomedical ontologies or WordNet [5] are utilized for the sources of semantic information. In the next section, we will introduce basic aspects of biomedical text mining and its resources for drug repositioning.

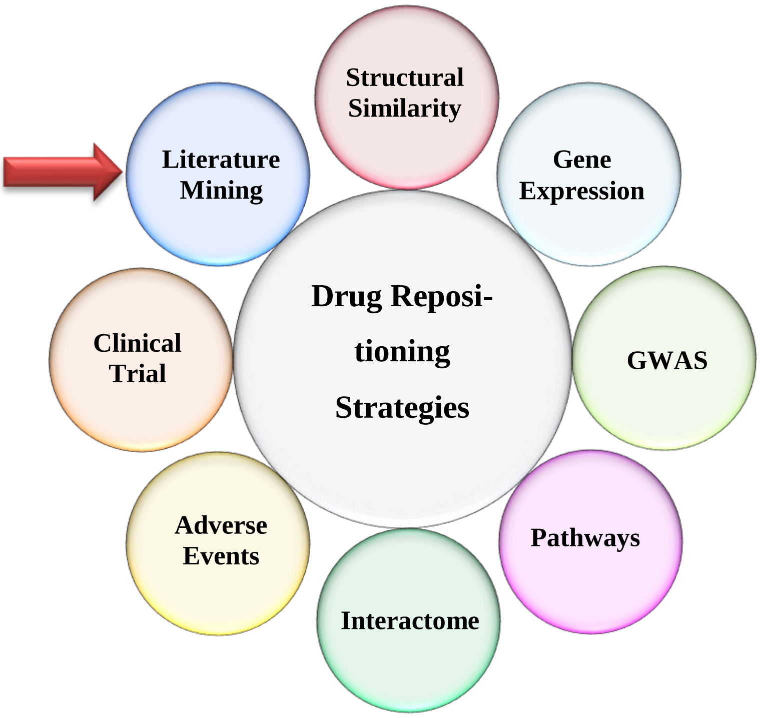


Figure 2.1 Main strategies of drug repositioning.

(Source: http://www.gvkbio.com/drugrepurposing/approach/repurposing-algorithms/)

2.2 Biomedical text mining

2.2.1 Basic concepts

Text mining

In general, text mining is the process of extracting information or discovering knowledge automatically from textual data or unstructured data. The process of text mining is quite similar to the process of data mining. However, the input of data mining systems is structured data, whereas text mining systems' input is unstructured data. Therefore, text mining needs to conduct the text preprocessing step in order to transform textual data into numerical data or structured data for further processing steps. Text mining is also known as an interdisciplinary area since it is related to various fields as shown in figure 2.2.

Biomedical text mining

Biomedical text mining refers to the application of text mining to biomedical domain. Through biomedical text mining, useful knowledge, which is hidden in biomedical literature, can be extracted. For example, named biomedical entities (i.e. genes, diseases, drugs, etc.) in texts are recognized and relations among them are also extracted.

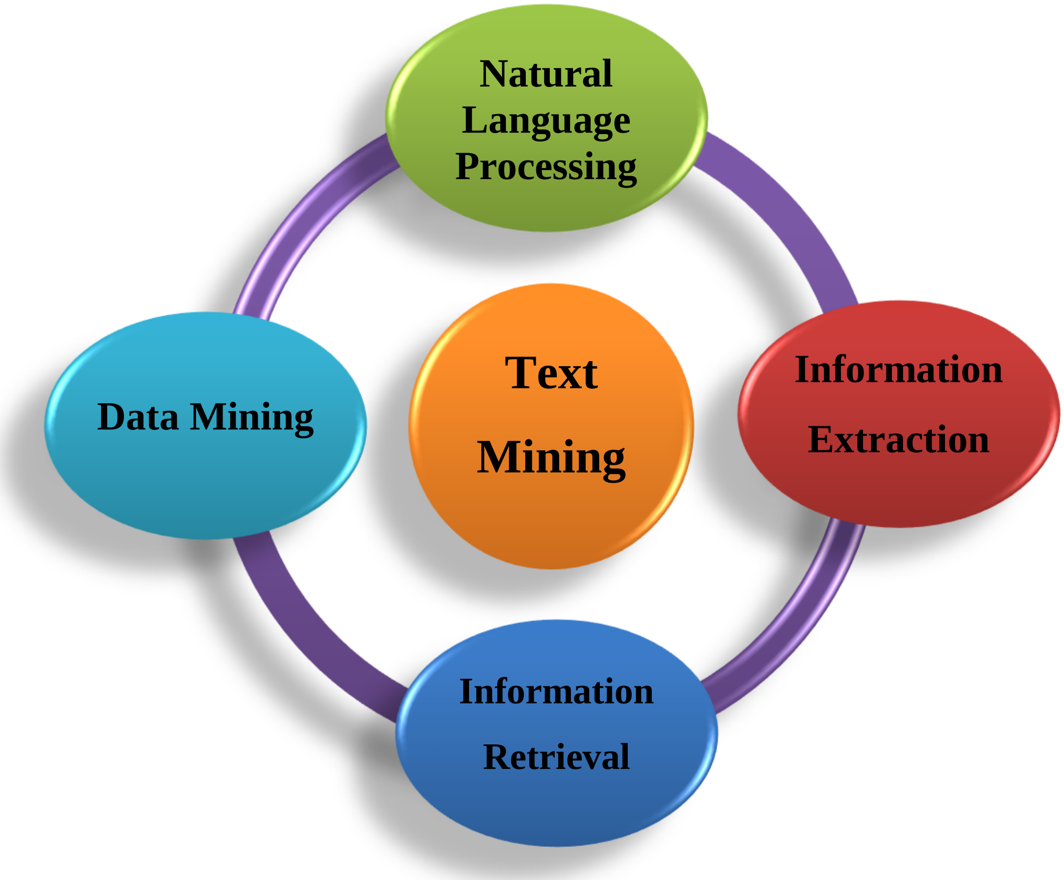


Figure 2.2 Related fields of text mining.

2.2.2 The process of text mining

In general, the process of text mining can be divided into five main phases, include:

- ✓ **Text gathering**
    - The process of text gathering is an important step for any text mining system because it affects directly the quality of text mining systems.

The crucial goal of text mining systems is to extract useful information from collected texts which called corpus. To meet this goal, therefore, we need to collect appropriate text resources.

- Choice of text resources for text mining systems depends on what kind of information we want to extract. For example, if we want to extract biomedical information, we can use PubMed abstracts as text resource.

✓ **Text preprocessing**

Text preprocessing includes the following tasks:

- ***Tokenization***

  Tokenization refers to the process which a text document is split into a set of words or terms (also called as tokens). However, before doing tokenization, special characters and punctuation need to be removed [9]. Words must be separated by white spaces.

- ***Removing stop words***

  Stop words typically refer to extremely frequent but meaningless words, for instance, articles (“a”, “an”, “the”), conjunctions (“and”, “or”, etc.), prepositions (“at”, “in”, “on”, etc.), and pronouns (“I”, “they”, “it”, etc.). These words should be removed in order to reduce the dimensionality of feature space. The most popular way to filter out stop words is based on an available list of stop words. It means that only words in the list will be filtered out from the corpus.

- ***Stemming***

  It is the process that the words of the same stem are reduced to their root or base form since they have equal or similar meaning. For example, the plural nouns will be changed to their singular forms; the "ing" or "ed" verbs will also be transformed into bare verbs [10]. One of the most widely used algorithms for stemming is Porter’s algorithm [11].

✓ **Feature generation**

In this step, we transform unstructured text data into structured data for further processing steps. In other words, we need to change text data into numerical data which is more appropriate to data mining algorithms. There are several ways of text representation such as Boolean model [12], Bag-of-Word model [13], and Word embedding model [7].

✓ **Feature selection and/or dimension reduction**

In case that input data, represented as a set of feature vectors, contain some features irrelevant from class label or objective variable, they often cause low performance of data mining, text mining, or machine learning. Feature selection is a popular method of increasing the performance.

Another serious problem about feature is so called “the curse of dimensionality”. Feature vectors consist of many dimensions easily cause long computation time, memory space exhaustion, and low performance in mining and learning. To solve this problem, dimension reduction is commonly conducted. Feature selection can be regarded as a kind of dimension reduction, however, there are other methods which compress high-dimensional feature vectors into low-dimensional ones. Traditional algorithms for dimension reduction or compression like Principal Component Analysis (PCA) and Latent Semantic Analysis (LSA).

Since most of text representation methods yield high-dimensional feature vectors, it is necessary to conduct feature selection and/or dimension reduction to reduce the dimensionalities of the representation of documents. It helps the further process of data mining to become more effective.

✓ **Data mining**

The most important purpose of text mining systems is to discover useful and unknown information from collection of text documents by using machine learning or data mining techniques. Therefore, it can be said that this step plays an important role in the process of text mining.

✓ **Analyzing and evaluating results**

This stage is to examine, analyze and evaluate obtained results from the process of data mining. From this, we can understand how well text mining systems work.

2.2.3 Approaches for text representation

Before applying data mining techniques to text documents, these documents need to be transformed into structured data or computable data. Some models of proposed text representation are listed as follows:

- ✓ **Boolean model**

  This model represents a document as a set of words or terms. If a term is present in a document, its weight is assigned value 1 (otherwise, value 0). Therefore, these terms' weights are all binary values (0 or 1). From this, each corpus is represented by a two-dimension matrix. Columns of the matrix correspond to all terms (vocabulary or dictionary) of corpus and rows correspond to the documents of corpus.

  For example, Table 2.1 shows a set of documents represented by Boolean model. Term weights are binary values (0 or 1) which mean if a term occurs in a document or not.

Table 2.1 Text representation by Boolean model.

	word 1	**word 2**	**word 3**	**...**
document 1	0	1	1	...
document 2	1	0	0	...
document 3	1	1	0	...
...	...	...	...	...

  It can be seen that this model has some drawbacks. Firstly, data space becomes extremely high dimensional and sparse. Secondly, since the importance level of terms is the same, it is hard to do feature selection.

- ✓ **Vector space model**

  Vector space model, which is also known as bag-of-word model, was proposed by Salton [13]. Since this model is quite simple and effective, it becomes one of the most popular models for text representation. Unlike Boolean model, in this model a term's weight is a real value which denotes its degree of importance in a document. There are various ways to compute term weights. Among them, term frequency (tf) is the simplest weight which indicates how many times a term presents in a document. In addition, Term Frequency-Inverse Document Frequency (tf.idf) is widely used for the term weight. This term weight was defined by Salton [14].

For example, table 2.2 show vector space model which is used to represent a set of documents. Term weights are the frequency of corresponding terms.

Table 2.2 Text representation by vector space model.

	word 1	**word 2**	**word 3**	**...**
document 1	0	6	2	...
document 2	1	0	0	...
document 3	4	3	0	...
...	...	...	...	...

Although this model has considered the importance level of terms in a document, it still has some limitations which do not make it become an effective model for text representation. One serious problem of both Boolean and vector space models is that they can make the data space become extremely high dimensional and sparse (containing so many zero values). Additionally, in this method the order and context of words in a sentence or are not considered.

- ✓ **Word embedding model**

To avoid limitations of two above models, Tomas Mikolov et al. [7] proposed word embedding model which is actively studied in these years. This model uses neural network to find a distributed representation of a word that is a short numerical vector (for example, 100 dimensions). This numerical vector is also called as a word vector. As a result of applying word embedding model, a set of word vectors (also known as word vector space) is generated by the methods called Continuous Bag of Words (CBOW) or Skip-gram. The former uses the context to predict a target word. In contrast, the latter uses a current word to predict surrounded words or the context of a word. These algorithms are simple Neural Networks algorithms with only three layers as shown in Figure 2.3

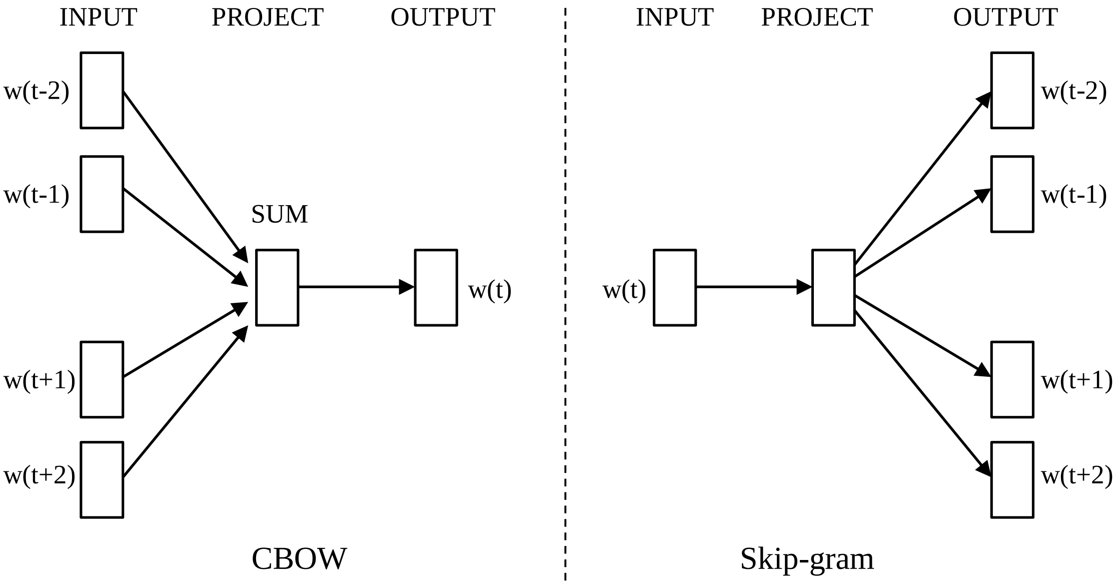


Figure 2.3 CBOW model and Skip-gram model.

(Source: Tomas Mikolov 2013, ICLR Workshop)

Unlike the previous models such as Boolean and vector space models, which only consider individual words for building the vector space, this model utilizes the context of each word to compute its vector representation. Therefore, the obtained vector space well represents the sense of words, that is, distance between two word vectors reflects the dissimilarity of two word senses. In addition, it is believed that, for various application domains, word analogy works well in this vector space.

In a sense, word analogy is inference of relation. Suppose that we have an input of three words “Tokyo”, “Japan”, and “Vietnam”. Here, “Japan” and “Tokyo” represents a given relation, and “Vietnam” is a part of new relation. As a result of word analogy, the output “Hanoi” is returned as the word nearest to the calculated point *X*, that is, *Tokyo* - *Japan* + *Vietnam*. In figure 2.4, the three words are located in the space of word vector generated by word embedding. Then, the displacement vector between the first two words is calculated. Using the displacement vector and the vector of the third word, the point *X* is calculated. Finally, “Hanoi” is returned as the nearest word to *X*.

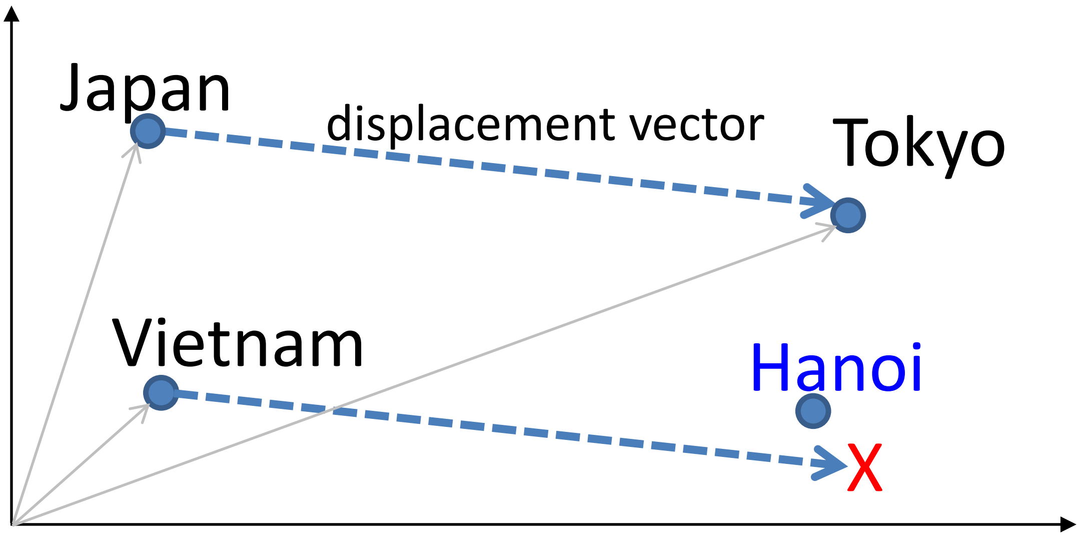


Figure 2.4 An example of word analogy.

2.2.4 Similarity measures

Briefly saying, similarity measures (or distance measures) are functions which compute the level of similarity between two vectors (or objects) in the n-dimension space. For example, suppose that we have two documents represented as two vectors, similarity measure between two vectors means the similarity between two corresponding documents.

It can be said that similarity measure plays an important role for clustering algorithms since it will affect directly the performance of clustering. Thus, different clustering algorithms can use different measures based on characteristics of data and application fields. Each measure has different characteristics that can be appropriate to the research problem or not.

The following metrics are widely used to calculate the similarity between objects or vectors in data mining algorithms.

- ✓ **Euclidean distance**

  For geometrical problems, Euclidean distance is a standard measure. In case of text mining, suppose that two term vectors $\overrightarrow{t_a}$ and $\overrightarrow{t_b}$ represent two given documents d_a and d_b respectively. The Euclidean distance between the two documents is calculated by following equation:

$$D_E(\overrightarrow{t_a}, \overrightarrow{t_b}) = \sqrt{\sum_{t=1}^{m} \left|w_{t,a} - w_{t,b}\right|^2} \qquad (Eq. 2.1)$$

where the term set is $T = \{t_1, \ldots, t_m\}$; $w_{t,a}$ and $w_{t,b}$ are term a and b weights, respectively.

- ✓ **Cosine similarity**

  Cosine similarity is one of widely-used similarity measures, especially in text mining. In this study, we also use this cosine similarity for our clustering algorithms. Eq. 2.2 shows the formula of cosine similarity.

$$SIM_C(\overrightarrow{t_a}, \overrightarrow{t_b}) = \frac{\overrightarrow{t_a}.\overrightarrow{t_b}}{\left|\overrightarrow{t_a}\right| \times \left|\overrightarrow{t_b}\right|} \qquad (Eq. 2.2)$$

- ✓ **Pearson's correlation coefficient**

  Like cosine similarity, Pearson's correlation coefficient also illustrates the correlation of two term vectors. Although the Pearson's correlation coefficient formula can be represented by different forms, it has the same meaning. A widely used form is

$$SIM_P(\overrightarrow{t_a}, \overrightarrow{t_b}) = \frac{m \sum_{t=1}^{m} w_{t,a} \,.\, w_{t,b} - TF_a.TF_b}{\sqrt{\left[m \sum_{t=1}^{m} w_{t,a}^2 - TF_a^2\right]\left[m \sum_{t=1}^{m} w_{t,b}^2 - TF_b^2\right]}} \qquad (Eq. 2.3)$$

where $TF_a = \sum_{t=1}^{m} w_{t,a}$ and $TF_b = \sum_{t=1}^{m} w_{t,b}$.

2.2.5 *Text mining algorithms*

Text classification

Text classification is a popular and fundamental problem in text mining. There are different methods or algorithms for this problem. Here, we present only two algorithms, Support Vector Machines (SVMs) and Neural Networks (NNs), which are broadly used for text classification.

- ✓ ***Support Vector Machines (SVMs)***

SVM was proposed by Vapnik et al. and widely used in data mining community [15]. It is one of the most effective machine learning algorithms for data classification in general and text classification in particular. According to [16], SVMs outperforms substantially and significantly existing methods on classification tasks. Especially, since SVMs can work well with high dimensional and sparse data spaces, it becomes an effective algorithm for text classification.

Additionally, SVMs can be combined with kernel methods to deal with non-linear classification problems. The main idea of using kernel function approaches in SVMs is to transform original non-linear feature space into new and linear one. In this learning model, we can use various kernels for complex classification problems.

- ✓ ***Artificial Neural Networks***

Artificial Neural Networks (ANNs), shortly called Neural Networks (NNs), are models which process information in a manner similar to the human brain. They are built up from a large number of units called as processing units or neurons, which are connected via links with linking weights. These neurons are unified and work together in order to address a certain problem in real life. Different NNs are designed for different applications (e.g. pattern recognition, classification, and clustering) through the learning process from training datasets. The main idea of the learning process is to modify or update the linking weights between neurons.

There are many different NNs. Among them, Perceptron is the simplest NN containing only one neuron. As shown in Figure 2.5, the input nodes receive the real values (x_i) and the output nodes produce value +1 or -1. The output of network is calculated by applying activation functions to values of the input vector along with their weights. After that, the output value u is combined with threshold *b* to determine the categorization values. Such a network can solve linear classification problems.

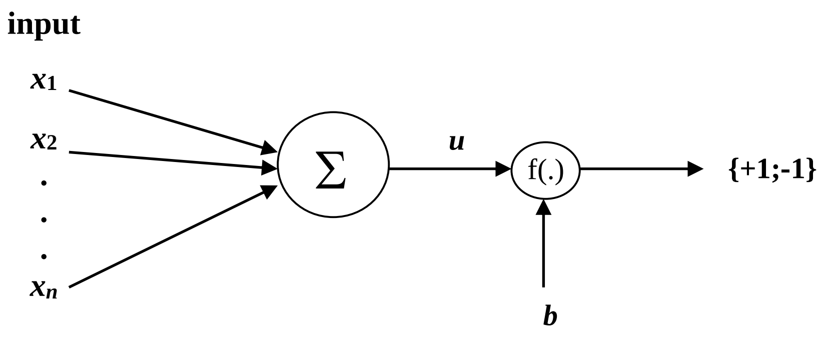


Figure 2.5 Perceptron as the simplest neural network.

However, more complex networks containing one or more hidden layers between input and output layers are mostly designed for real-world problems (see Figure 2.6). Due to NNs only work with numerical data, all real-world data such as images, sound, text, or time series must be transformed into numerical vectors.

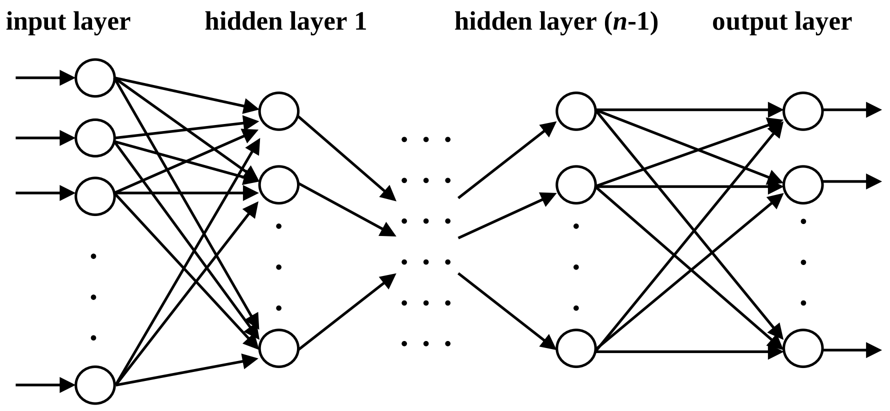


Figure 2.6 General architecture of the multi-layer neural network.

Text clustering

Also, text clustering is an important problem in text mining. It is used to find groups or clusters of documents (represented as vectors) having similar content. After clustering, all documents are partitioned into some clusters. The documents in the same cluster are more similar than those of other clusters.

In general, clustering results mainly depend on the distribution of the data and similarity measures used. However, to obtain an ideal clustering result, we should consider carefully different aspects of applications as well as users. There are various algorithms with different advantages and disadvantages. Here we introduce some widely-used clustering algorithms and standard evaluation methods for the text clustering problem.

✓ ***Algorithms for clustering***

- **Partitioning clustering algorithms**

  These clustering techniques attempt to partition a dataset into *K* clusters by optimizing a given criterion. First, they select k random data points considered as k centroids of k clusters. Next, they assign data points to the cluster of the nearest centroids. After that, all centroids will be updated. This process repeats until no assignment occurs or a given condition is satisfied. Simple examples of partitioning clustering algorithms are *k*-means, *k*-medoids [17], and PAM [18]. These algorithms are widely used in practical applications. However, one of disadvantages is that we need to first determine the number of clusters *k*.

- **Hierarchical clustering algorithms**

  There are two opposite ways of doing hierarchical clustering [17]. In the first one, it starts with each data point belonging to one of the disjoint clusters then the two most similar clusters are merged together, one by one. In contrast, the other way starts with the whole dataset considered as only one cluster, then split them into two most different clusters. These processes continue until stop conditions are satisfied. For example, figure 2.7 shows the result of clustering the USArrests dataset by the first algorithm above. The tree in the figure is called dendrogram. For treating complex and non-spherical shape of cluster, *CURE* [19] is used.

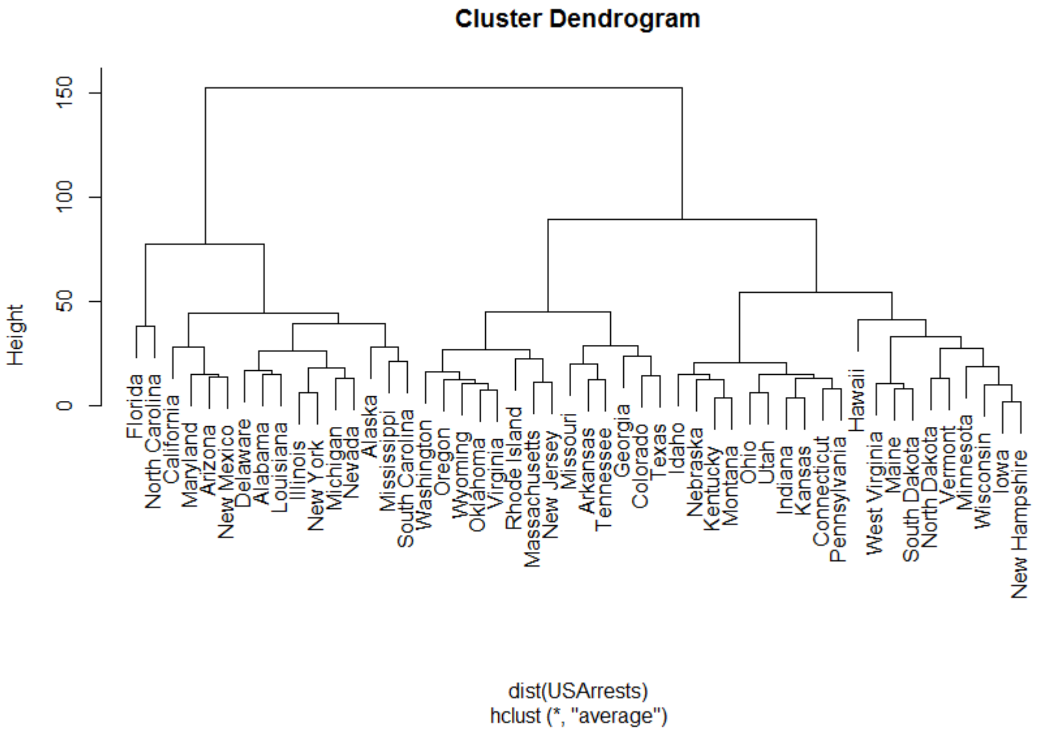


Figure 2.7 An example of dendrogram.

Information extraction (IE)

Information extraction is the basic task for text mining. In some cases, IE can be considered as the process of feature generation for further processing of text mining. In other words, IE can be conducted to build the feature space for text mining. The process of information extraction includes two main sub-tasks as follows:

✓ **Named Entity Recognition (NER)**

The first step of information extraction is NER which is used to identify named entities mentioned in texts. Named entities include the name of persons, locations, genes, diseases, drugs, chemical compounds, etc. Typically, NER is conducted in three steps: determining the boundaries of an entity within the text, classifying this entity into a predefined class, and normalizing this entity to a standard name in specific domains such as biology, medicine, etc.

To extract named entities in texts, we can use one of the following methods or combine these methods together.

- *Dictionary-based methods*

  The Dictionary-based method is the simplest approach which uses a prepared dictionary of terms (represented as entities) to determine if an entity occurs in texts or not. In such a system, every word or group of words of the text is matched terms of biomedical dictionary (see Figure 2.8). If a term is matched in the dictionary, it is tagged. The precision of these methods is generally high, but its recall is poor due to the existence of spelling mistakes and morphological variants [20]. Furthermore, homonymy can be another reason for decreasing precision [21].

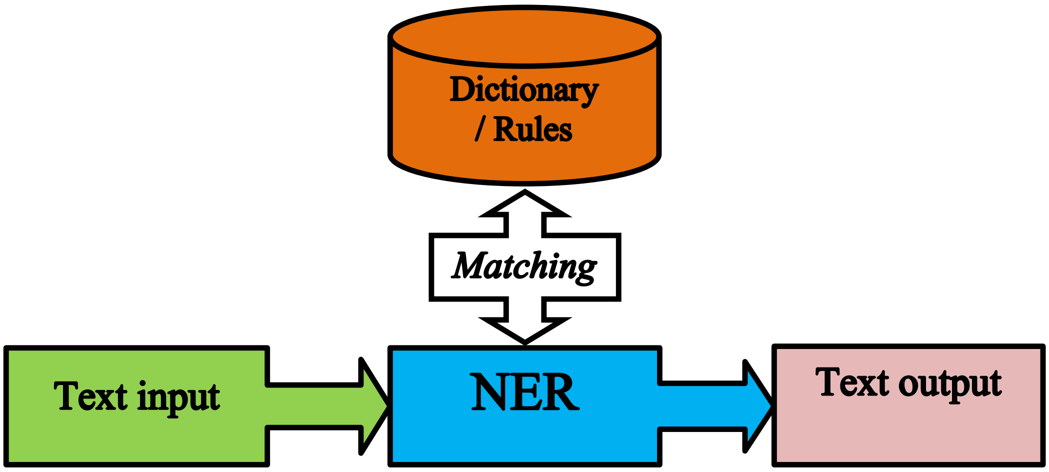


Figure 2.8 The process of the rule-based or dictionary-based approach.

- *Rule-based methods*

  This method uses heuristic rules for recognizing entity names in texts (see Figure 2.8). Depending on the characteristic of different entity types, we can create the suitable rules for entity recognition. Each entity has different characteristics such as the first capital letter of a word, prefixes and suffixes of a word, the preceding-word of a word, etc. However, it is hard to design the rules which are appropriate for all cases.

- *Machine-learning-based methods*

  This method applies machine learning algorithms to automatically recognize entities by classifying terms from texts into predefined entity categories. Such a method often requires enough good training datasets to learning classifiers, however, it is hard to find such datasets in many applications.

- ✓ **Relation Extraction (RE)**

  After named entities have been recognized in texts, the relations between pairs of these entities need to be extracted. The relation extraction is important in various domains because it helps us discover useful information from available text repositories. Especially, in biomedical domain, it can be used to find new knowledge about genes, proteins, diseases, drugs, etc. The task of relation extraction is to identify relationships between pairs of recognized entities in a given literature resource such as Reuters-News, PubMed corpora, etc.

  Relation extraction methods are described from simple to complex ones, as follows:

  - *Co-occurrence-based methods*

    These are the simplest methods with assumption that if two entities co-occur in sentences, paragraphs, or documents many times, they are likely to have some relationships together. Therefore, the main goal of these approaches is to identify if a pair of given entities appears in a piece of text or not. This is the simplest way to extract relations between entities.

  - *Rule-based methods*

    These methods utilize the linguistic rules (also called the patterns) as clues to identify particular relations. To generate these rules, we require the support from domain or linguistic experts.

  - *Machine-learning-based methods*

    These methods often rely on supervised machine learning techniques in order to automatically identify relations in texts. However, they typically require available training data which consists of given relations for learning classifiers.

Chapter 3

Materials and Methods

In this chapter, we would like to describe about materials and approaches which used in the process of data preparation for biomedical information extraction. In this study, cancer-related PubMed abstracts and other biomedical ontologies are used as the input of the experimental system. Additionally, some algorithms, which are applied to process data, are also presented. Especially, a strategy is also proposed to solve the class imbalance problem.

3.1 Overview of processing pipeline

The processing pipeline in this study is shown in Figure 3.1. It includes four phases as follows:

- **Phase 1** - Text processing;
- **Phase 2** - Word embedding;
- **Phase 3** - Combination of word vectors and database information;
- **Phase 4** - Generation of drug-disease relation vectors.

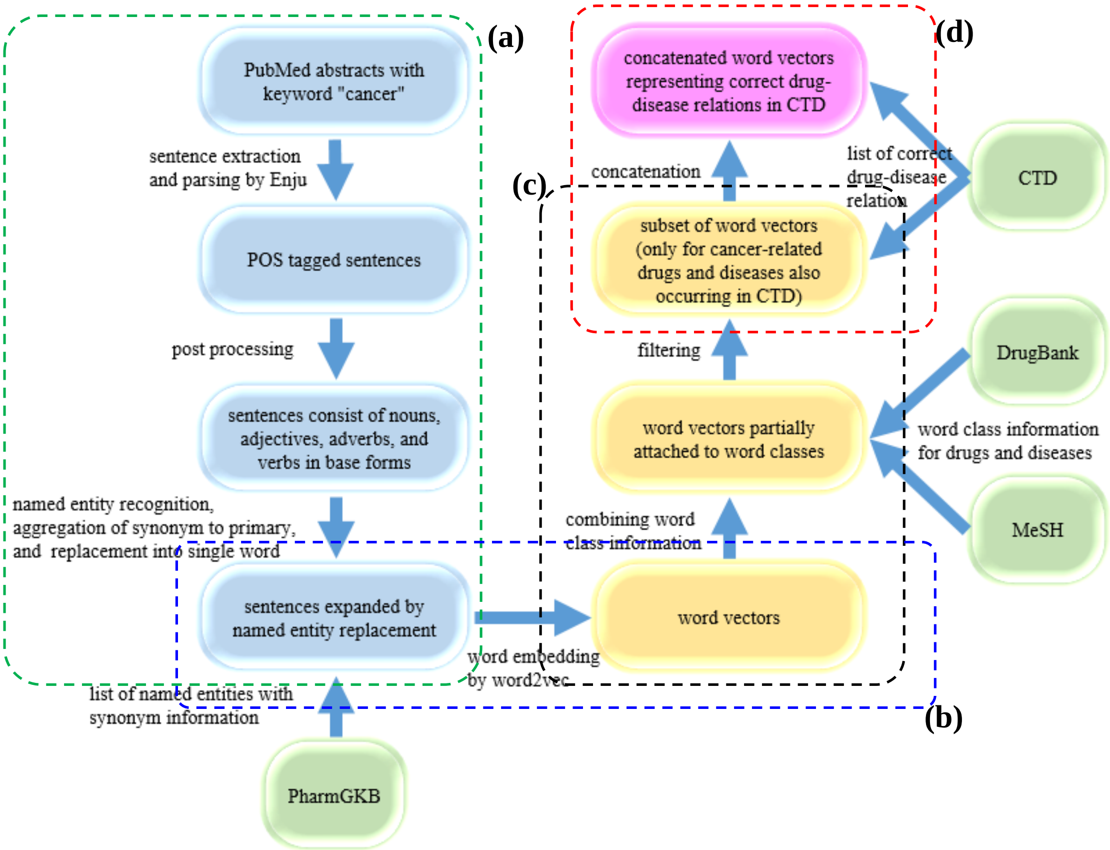

(a)
PubMed abstracts with keyword "cancer"
sentence extraction and parsing by Enju
POS tagged sentences
post processing
sentences consist of nouns, adjectives, adverbs, and verbs in base forms
named entity recognition, aggregation of synonym to primary, and replacement into single word
sentences expanded by named entity replacement
list of named entities with synonym information
PharmGKB
word embedding by word2vec
(b)
word vectors
combining word class information
(c)
word vectors partially attached to word classes
word class information for drugs and diseases
DrugBank
MeSH
filtering
subset of word vectors (only for cancer-related drugs and diseases also occurring in CTD)
concatenation
(d)
concatenated word vectors representing correct drug-disease relations in CTD
list of correct drug-disease relation
CTD

Figure 3.1 Overview of processing pipeline. Box colors indicate: light blue for corpus, light green for databases, yellow for word vectors, and pink for concatenated word vectors. (a), (b), (c), and (d) corresponding to four phases of processing pipeline.

3.2 Resources and tools

3.2.1 Biomedical databases

- ✓ **PubMed**

PubMed is widely used in biomedical text mining community. It allows users to access to a large biomedical database including the summary information of published biomedical articles such as author, title, abstract, keywords, etc. Currently, this database contains more than 24 million references to biomedical journals.

We can access to the PubMed database via a web-based search engine ([22, 23]). In this study, we downloaded a subset of PubMed abstracts to extract biomedical information extraction related to drugs, diseases, and genes. Since we are interested in drug repositioning for cancer diseases, only the cancer-related abstracts were downloaded.

- ✓ **Medical Subject Headings (MeSH)**

MeSH is a comprehensive controlled vocabulary thesaurus of biomedical terms which is created and updated by the United States National Library of Medical (NLM). The biomedical abstract databases like MEDLINE and PubMed use the MeSH terms as a document indexing system. Most subject headings have a short description or definition. MeSH descriptors are arranged in both an alphabetic and a hierarchical tree. The tree locations contain systematic labels which are also known as tree numbers.

In this study, we combined disease word vectors with MeSH tree numbers. Since we are interested in cancer or cancer-related diseases which are classified in “C04”, only the tree numbers beginning with “C04” are concatenated with disease word vectors.

The MeSH database is yearly updated and is available online. We can download MeSH free of charge through the website of it [24] or we can also obtain it from the MeSH download page [25] in XML and other formats.

- ✓ **Pharmacogenomics Knowledgebase (PharmGKB)**

PharmGKB is a pharmacogenomics knowledge resource which is used to aggregate, curate, integrate, and disseminate information about the relationships between human genetic variations and corresponding drug responses [26]. At present, it is maintained at Stanford University. We can learn more detail about PharmGKB database and download from its web page [27].

In our research, we used PharmGKB as a dictionary for recognizing biological named entities. Additionally, it is used to transform synonyms of biomedical terms occurring in PubMed abstracts into their primary forms. This is useful for information extraction tasks such as biomedical named entity recognition and relation extraction.

- ✓ **DrugBank**

DrugBank is a pharmaceutical database which contains knowledge about drugs and drug targets. We can learn more about the DrugBank database in [28]. In DrugBank, Anatomical Therapeutic Chemical (ATC) code is used to classify active ingredients of drugs according to the system or organ. ATC code consists of five different levels from general level to detail level.

Since we are interested in drug repositioning about cancer, only ATC codes starting from L is used to build drug word vectors in our study. To get DrugBank database, we can access to the home page of DrugBank [29].

- ✓ **Comparative Toxicogenomics Database (CTD)**

CTD provides useful information regarding relationships between different types of biomedical objects such as chemicals, and genes, and diseases [30]. These relationships are manually extracted from literature by biomedical experts. Therefore, these relationships can be considered as a gold standard resource for biomedical applications. In scope of this study, we extracted drug-disease relations which are written in CTD. We can download the CTD database from the website [31].

3.2.2 Tools

- ✓ **Enju parser**

Enju is a software for parsing English sentences developed at Tsujii Laboratory in Tokyo University. As a result of parsing, Enju outputs phrase structures and predicate-argument structures which are useful for NLP applications such as information extraction, machine translation, etc.

The Enju parser has the following advantages:

- It is a deep parser.
- The parsing speed and accuracy are high.
- Specially, it can work well with large biomedical corpora.

In our study, Enju is used for Part-Of-Speech (POS) tagging and conversion of words into base forms. To learn more about the Enju parser, we can refer to the website [32].

✓ **word2vec software for word embedding**

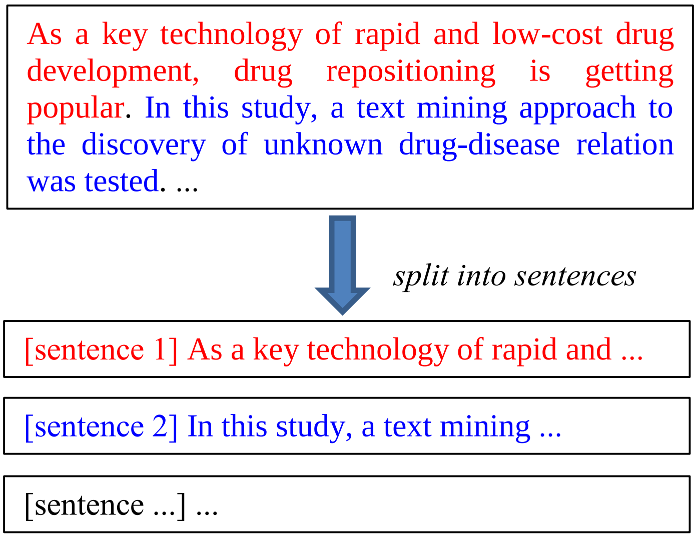
This tool provides an effective implementation for computing vector representations of words. It was developed by Tomas Mikolov and his co-workers. It is an open source software and for research purposes, it can be freely download at [33].

The input of word2vec is a set of texts (saved as plain text file) and its output is a set of word vectors (also saved as text file). Each vector represents a word or a term in texts. These word vectors can be used to find similar words since the vectors of similar words are likely to locate close to each other in vector space. In addition, word analogy also is a useful function of word2vec. In a sense, word analogy is inference of relation.

3.3 Data preparation

3.3.1 Cancer-related corpus from PubMed abstracts

As a raw corpus, we used a subset of PubMed abstracts downloaded in October 2013, filtered by the keyword “cancer”. From 3,099,076 abstracts, 14,847,050 sentences were extracted. Figure 3.2 shows an example of splitting an abstract into sentences.



As a key technology of rapid and low-cost drug development, drug repositioning is getting popular. In this study, a text mining approach to the discovery of unknown drug-disease relation was tested. ...

[sentence 1] As a key technology of rapid and and ...

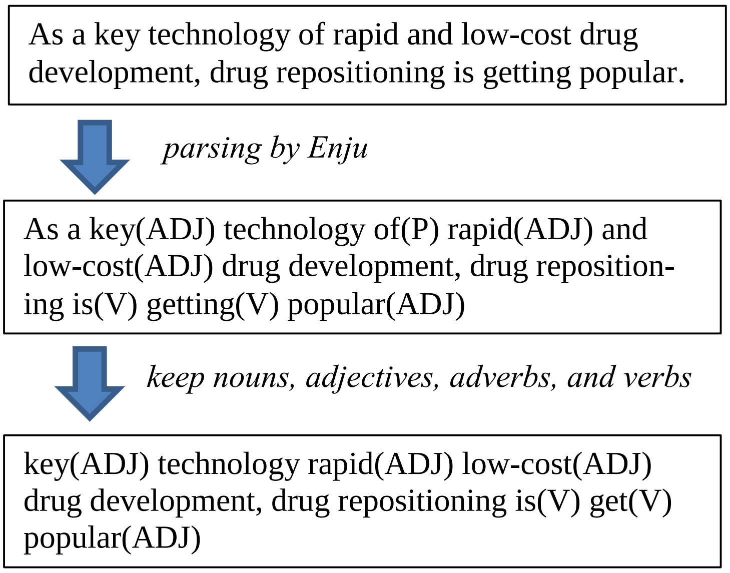
[sentence 2] In this study, a text mining ...

[sentence ...] ...

Figure 3.2 An example of splitting an abstract into sentences.

3.3.2 Parsing sentences

Enju [34] was used for POS recognition of words and conversion into base forms. Since the sentences were extracted from biomedical abstracts, "-genia" option was specified. As a result, POS and base form are recognized for each word.



As a key technology of rapid and low-cost drug development, drug repositioning is getting popular.

As a key(ADJ) technology of(P) rapid(ADJ) and low-cost(ADJ) drug development, drug repositioning is(V) getting(V) popular(ADJ)

key(ADJ) technology rapid(ADJ) low-cost(ADJ) drug development, drug repositioning is(V) get(V) popular(ADJ)

Figure 3.3 An example of POS tagging, base form conversion, and filtering by POS category.

So that word2vec can differently treat the same word with different POS categories, they were attached right after the base form of words (e.g. "care" -> "care(V)").

For readability, nouns are kept as is. To simplify the input for word2vec, we removed all words except nouns, adjectives, adverbs, and verbs as shown in Figure 3.3.

3.3.3 Named entity recognition and conversion into single words

Biological terms typically consist of two or more words. In addition, they have many synonyms. Since word2vec basically treats a sentence as a sequence of words, it is needed to recognize biological synonyms, aggregate them into primary terms, and convert them into single words. In this study, primary names and synonyms of drugs, diseases, and genes were extracted from PharmGKB [26] and used for recognition and aggregation (genes are used only for showing distribution of word vectors). For each converted single words, prefixes indicating their semantic categories were attached for later processing, as an example is shown in Figure 3.4.

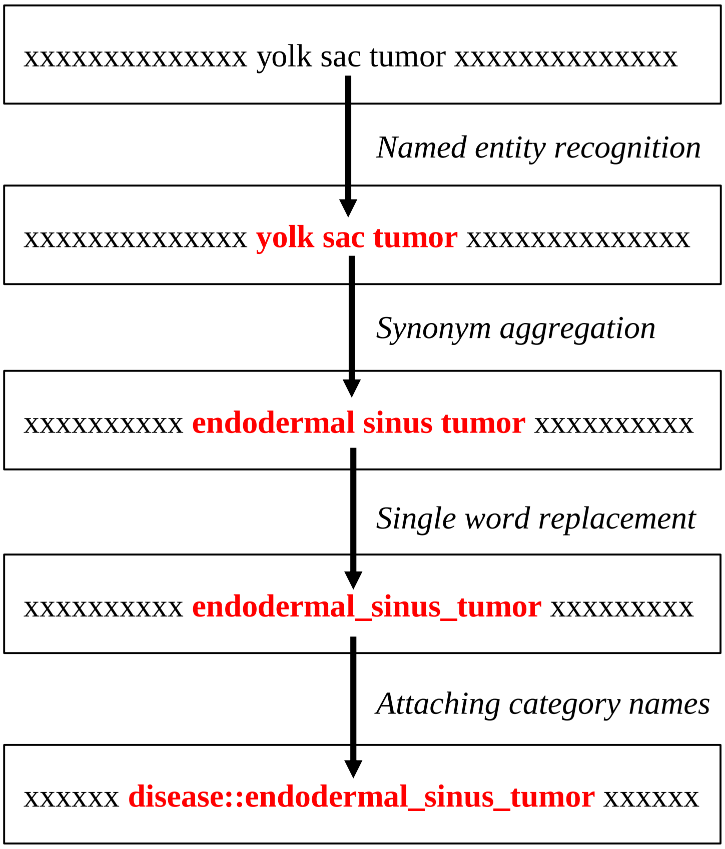

xxxxxxxxxxxxxx yolk sac tumor xxxxxxxxxxxxxx
Named entity recognition
xxxxxxxxxxxxxx yolk sac tumor xxxxxxxxxxxxxx
Synonym aggregation
xxxxxxxxxx endodermal sinus tumor xxxxxxxxxx
Single word replacement
xxxxxxxxxx endodermal_sinus_tumor xxxxxxxxx
Attaching category names
xxxxxx disease::endodermal_sinus_tumor xxxxxx

Figure 3.4 Single word replacement for term "yolk sac tumor" in a sentence.

Related to the conversion above, we need to consider about the existence of original single words. Firstly, if a synonym word is aggregated into primary word, the original word disappears and is not used for word embedding. Secondly, if a multi-word term is converted into a single word, all the original single words in the multi-word term disappear. Thirdly, if two multi-word terms occur in a sentence with overlapping, it is impossible to replace both of them at once. To avoid there problems, a sentence is converted into the sentences which containing at most one converted word per sentence. For example, if a sentence contains two terms to be converted, three sentences including original one are generated. After all conversion, 14,847,050 sentences are expanded to 45,264,480.

3.3.4 Word embedding

In the field of text mining and natural language processing, computational text representation of a linguistic unit (e.g. documents, paragraphs, sentences, terms, and words) is essential. The simplest one for document is bag-of-word model in which each document is represented as a vector of word frequencies. In case of word representation, only the neighboring words in the same sentence are counted. For better analysis, stop-words are removed and raw frequencies are modified by term weighting such as tf-idf. After that, these vectors are used to evaluate the characteristics of the units and similarities between them (vector space model).

One of the serious problems in such a representation and analysis is high dimensionality and sparseness of vectors. For instance, 10 millions of sentences may contain one million of different words, then the dimension of a vector is also one million. In addition, since frequency of word follows Zipf's law, most of the one million of words only occur a few times, which makes the vectors quite sparse. Though it is possible to reduce the number of dimensions by PCA or LSA, this problem is not fully solved.

Word embedding for distributed representation of word sense is a new approach to this problem. Based on neural network algorithm, reasonably short numerical vectors (e.g. 100 dimensions) are calculated for all words in a set of sentences. Through the application studies, it is proved that the vector space constructed by word embedding represents word senses and distances (similarities) between them quite well. Additionally, in this space of word sense, word analogy works well in some domains. For example, given three words “man”, “woman”, and “king”, word analogy could predict "queen" by calculating *vector("man") - vector("woman") + vector("king")* and searching for the nearest word vector *vector("queen")*. Though word analogy might allow wide variety of applications, the most desired one is discovery of unknown relations.

In this study, we used word2vec software, a de facto standard implementation of word embedding algorithm, with the following parameters by default.

- vector size = 200
- window size = 8
- minimum count of words to be embedded = 1 (i.e. all words)
- model = continuous bag of words

As a result, 1,772,186 words were embedded into word vectors (2,303 for drugs, 3,069 for diseases, 8,703 for genes, and 1,758,111 for others).

3.3.5 Combination of word vectors and database information

For the evaluation of clustering results, ATC codes [35] and MeSH tree numbers [36] were attached to drug and disease names, respectively. ATC codes were extracted from DrugBank [28]. Due to the incompleteness of data annotation, only 1,253 drugs out of 2,303 and 2,745 diseases out of 3,069 have such classification.

ATC code has hierarchical structure. For example, the ATC code "L01XE02" attached to "drug::gefitinib" can be interpreted as follows:

L: ANTINEOPLASTIC AND IMMUNOMODULATING AGENTS
L01: ANTINEOPLASTIC AGENTS
L01X: OTHER ANTINEOPLASTIC AGENTS
L01XE: Protein kinase inhibitors
L01XE02: gefitinib

Since we are interested in drug repositioning about cancer, ATC codes must start from "L".

For disease name, MeSH tree number is available and it also has structured code. For instance, the MeSH tree number "C04.557.470.200.025.540" is attached to "disease:klatskin_tumour", and its first four digits are interpreted as follows:

C: DISEASES
C04: Neoplasms
C04.557: Neoplasms by Histologic Type
C04.557.470: Neoplasms, Glandular and Epithelial
C04.557.470.200: Carcinoma
C04.557.470.200.025: Adenocarcinoma

In this system, cancer or cancer-related diseases are clearly marked by the first digit "C04".

3.3.6 Generation of the special vector for drug-disease relations

For the evaluation of difference vectors between drugs and diseases, relations between drugs and diseases occurring in the corpus were extracted from CTD [30]. The figure 3.5 illustrates the concatenation of drug and disease vectors to create drug-disease relation vectors. If the relation is written in CTD with therapeutic evidence, it is labeled "TRUE" (otherwise, "FALSE"). Only the 12,462 relations with therapeutic evidences were adopted for obtaining trustable results. In the set of relations, the mapping from drugs to diseases is many-to-many. For example, "drug::gefitinib" is related to 17 different diseases, and "disease::lung_neoplasm" is mapped from 60 different drugs.

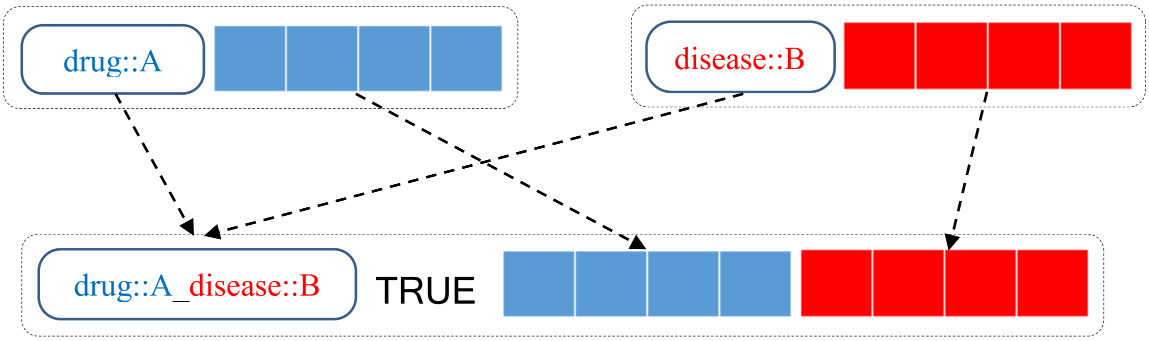


Figure 3.5 The concatenation of drug and disease word vectors to create drug-disease relation vector. If the relation is written in CTD with therapeutic evidence, it is labeled "TRUE" (otherwise, "FALSE").

In order to conduct detailed analysis on cancer-related drugs and diseases, 12,462 extracted drug-disease relations were further filtered so that both of drug and disease names in each relation are attached to an ATC code and a MeSH tree number beginning with “L” (Antineoplastic and immunomodulating agents) and “C04” (Neoplasms), respectively. As a result, 1,097 relations consist of 104 anti-cancer drugs and 107 cancer-related diseases were extracted for detailed analysis.

3.4 Algorithms

3.4.1 Checking the distribution of biomedical word vectors

In order to check the distribution of biomedical word vectors for drugs, diseases, and genes, we adopted PCA to convert 200-dimension vectors into 3-dimension vectors and plotted in 3D space. In this study, prcomp and plot3d functions included in stats and rgl packages respectively for R software were used.

3.4.2 Clustering for drugs and diseases

For visual evaluation of word vector quality, we performed hierarchical clustering with cosine distance and Ward's method [37]. Before the clustering, 2,303 drugs and 3,069 diseases occurring in the corpus were reduced to 1,282 and 1,051, respectively, since other drugs and diseases did not occur in CTD. In this study, we used hclust and plot functions included in stats and graphics packages respectively in R software for hierarchical clustering and visualizing clustering performance.

3.4.3 Classification for drug-disease relations

SVM was adopted for learning and predicting possible relations between drugs and diseases. As an implementation, ksvm function included in kernlab package for R software was used with default parameters.

3.4.4 Cross-validation

Cross-validation is a popular method which is used to evaluate the performance in classification problems. In this study, we used 10-fold cross-validation method to evaluate the classification performance of unseen drug-disease relations. In this method, a given dataset is randomly divided into 10 equal folds. After that, the training and testing process is conducted 10 times by choosing alternately 1 fold for testing and the other 9 folds for training. To avoid the effects of random division on the classification performance, the 10-fold cross-validation was performed 100 times, and the accuracies were averaged.

3.4.5 Methods for the class-imbalance problem

The class-imbalance problem is also called imbalanced dataset problem. A dataset is considered class-imbalanced if the number of examples in some classes is significantly larger than in other classes. In case of two-class datasets, *the minority class* contains small amount of examples (also called *positive examples*), while *the majority class* consists of large amount of examples (also called *negative examples*). Such imbalanced datasets are often found in various classification problems, especially in bioinformatics. When we apply standard machine learning algorithms to imbalanced datasets, it often gives a poor performance. Therefore, we need to find the way to deal with this problem.

To solve this problem, many strategies have been proposed. In general, they can be divided into two groups of methods: data level methods and algorithmic level methods. In this study, we applied the data level method in which corresponding to the number of positive examples (or *correct drug-disease relations*), the same number of negative examples (or *incorrect drug-disease relations*) were randomly selected, and used in each cross-validation as shown in Figure 3.6.

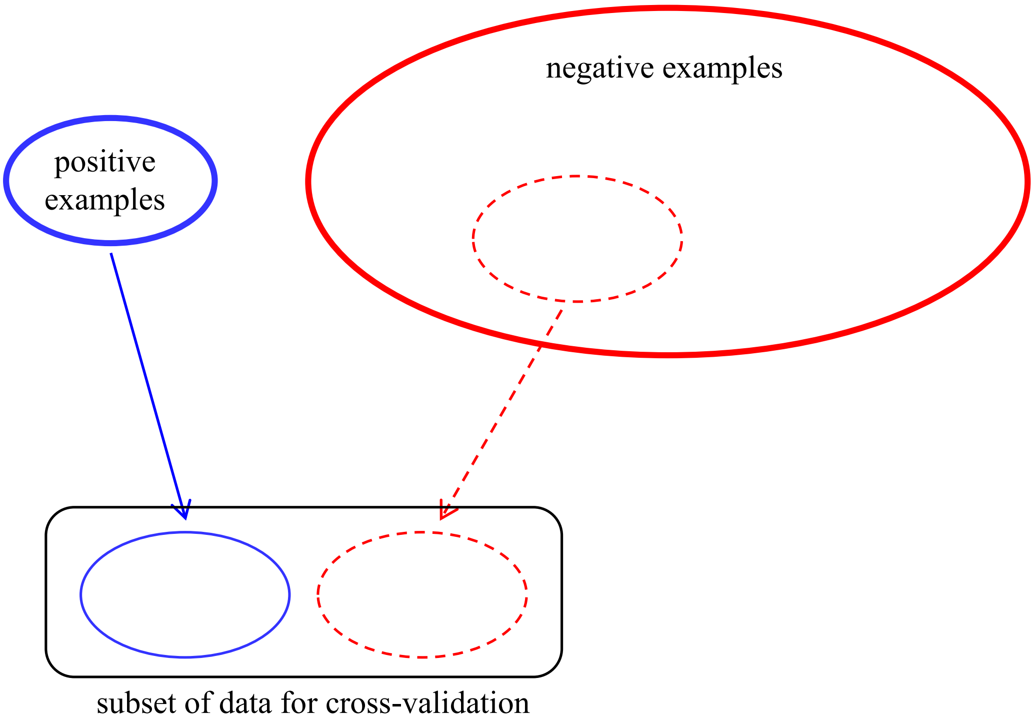


Figure 3.6 A strategy for solving the class imbalance problem.

Chapter 4

Experimental Results and Discussion

This chapter aims to illustrate the result of plotting and clustering word vectors for drugs and diseases. Additionally, the classification performance of drug-disease relation vectors is presented. We also discuss on the experimental result for explaining the reason why this result is good or not good. Though further screening based on experts' knowledge is necessary, this result demonstrates that the classification of concatenated word vector is a promising approach to in-silico screening of drug-disease relations for drug repositioning.

4.1 Distribution of drug-disease-gene vectors

Figure 4.1 illustrates the 3D plot of vectors corresponding to 2,303 drugs, 3,069 diseases, and 8,703 genes. For visualization, the dimension of vector was reduced from 200 to 3 by PCA. In the top panel of the figure, it is shown that the distributions of word vectors in three categories are clearly separated. In the bottom panel, it is also shown that the frequent words have clear separation, whereas it is relatively difficult to discriminate the categories of rare words.

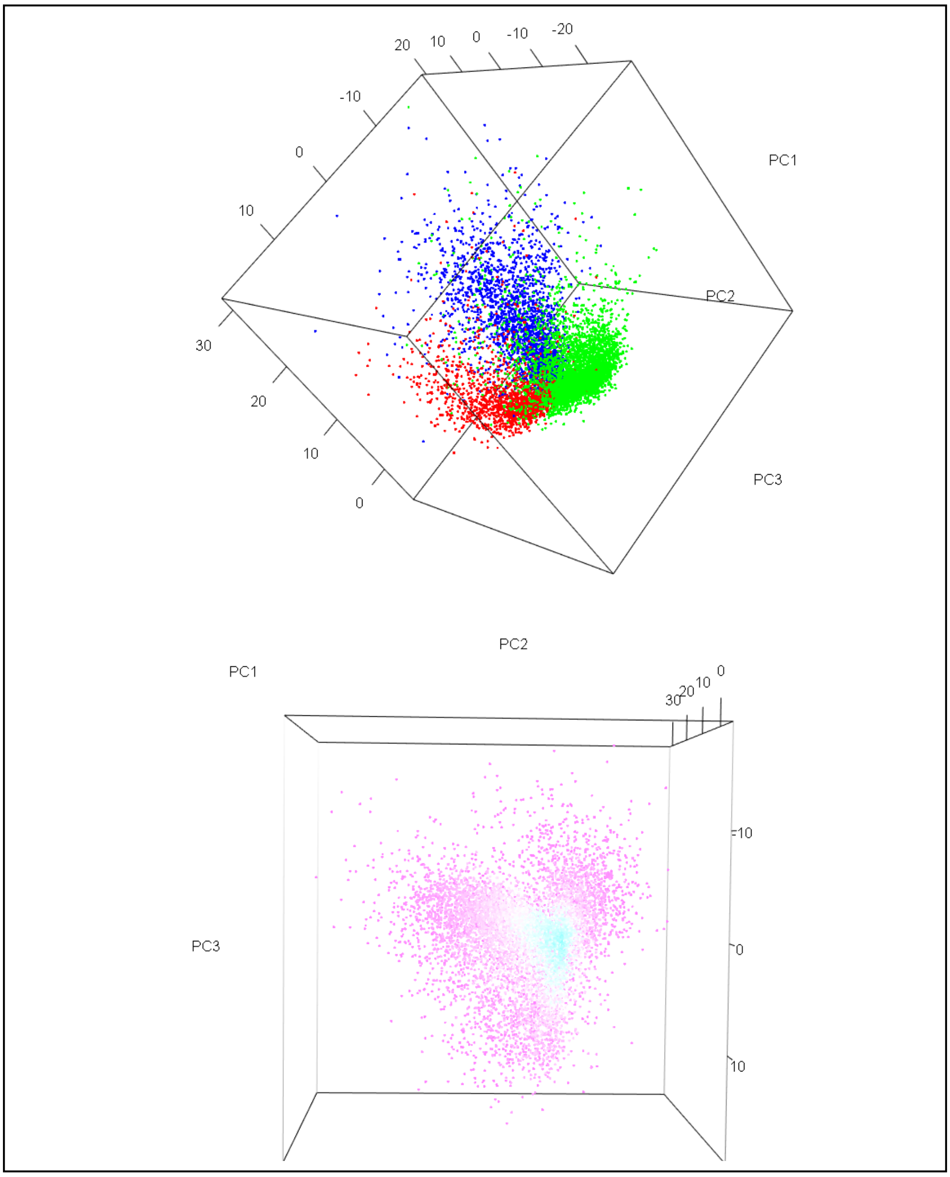


Figure 4.1 Distribution of word vectors visualized through PCA and 3D plot. Top panel: blue, red, green colors indicate word vectors for drugs, diseases, and genes. Bottom panel: color gradation from light blue to light pink indicates the frequency of words (from rare to frequent).

4.2 Cluster analysis for drug and disease vectors

Figures 4.2 and 4.3 show the results of hierarchical clustering for drugs and diseases, respectively. In the right panels of them, entire pictures of clustering results for 1,282 drugs and 1,051 diseases are shown. In the right panel of Figure 4.2, it can be seen that most of the anti-cancer drugs are condensed in the ninth cluster from the top (left panel for more detail). It indicates that the word vectors for drugs well represent the characteristics of corresponding drugs. Also in Figure 4.3, we can see that the seventh cluster from the top contains a number of cancer-related diseases, however, also in sixth and ninth clusters. The difference between these results might come from the fact that diseases can be classified from different perspectives (tissues, mechanism, etc.).

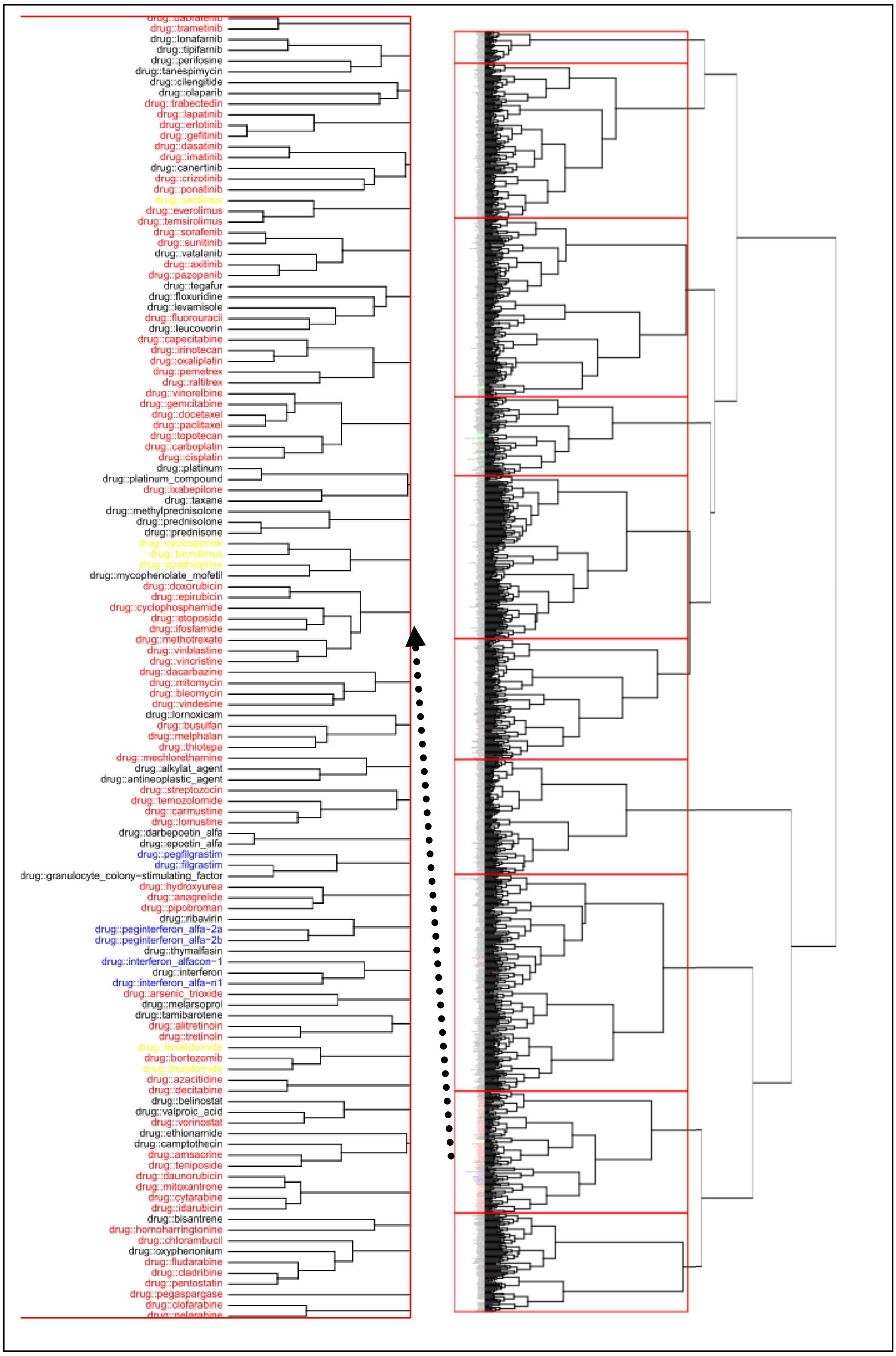


Figure 4.2 Result of hierarchical clustering on drugs. Red, green, blue, yellow colors for characters indicate that the drugs are classified in ATC codes as “L01:Antineoplastic Agents”, “L02:Endocrine Therapy”, “L03:Immunostimulants”, and “L04:Immunosuppressants”, respectively.

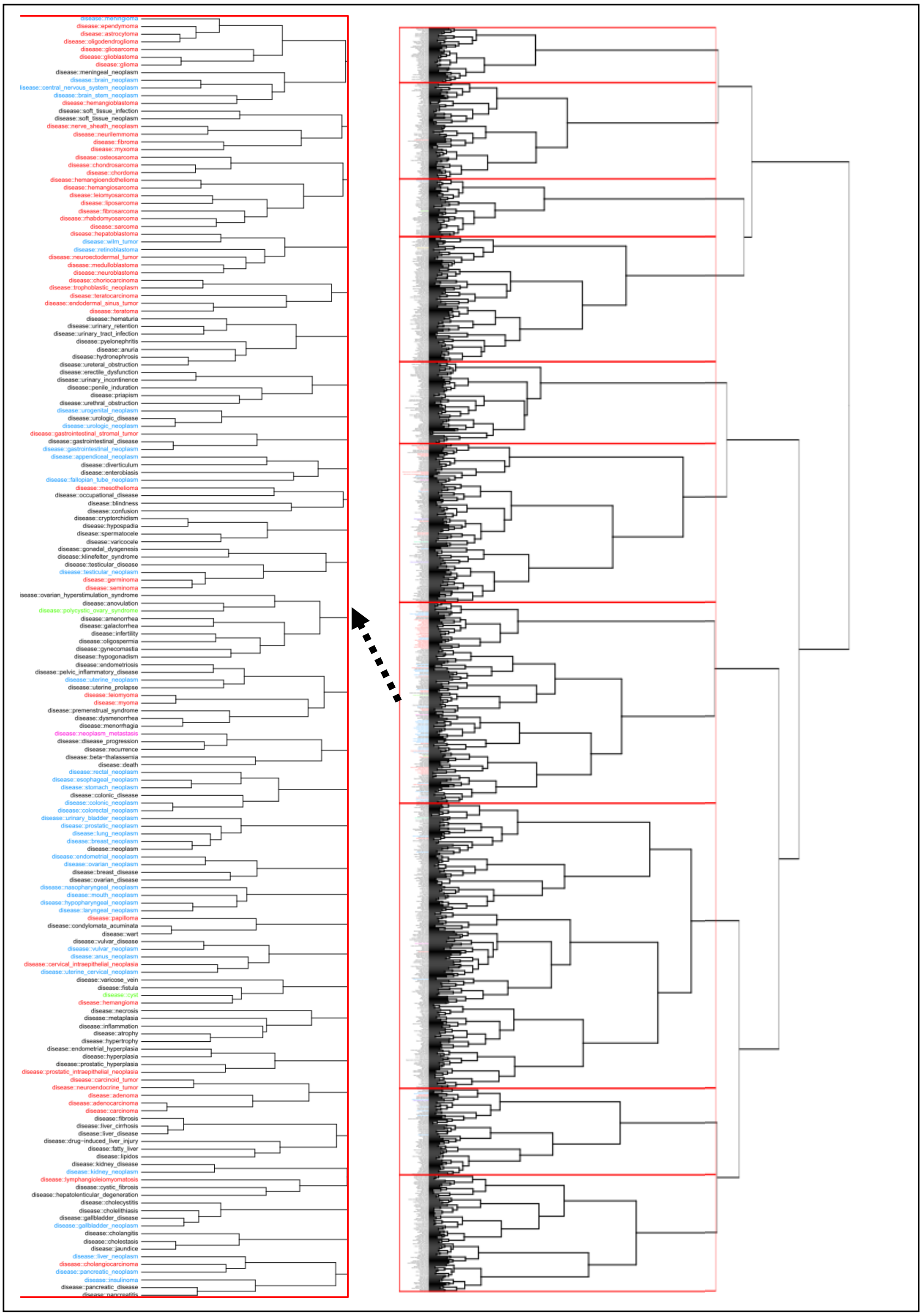


Figure 4.3 Result of hierarchical clustering on diseases. Green, yellow, red, dodgerblue, pink, blue, and springgreen colors for characters indicate that the diseases are classified in MeSH tree numbers as "C04.182:Cysts", "C04.445:Hamartoma", "C04.557:Neoplasms by Histologic Type", "C04.588:Neoplasms by Site", "C04.697:Neoplastic Processes", "C04.730:Paraneoplastic Syndromes", and "C04.834:Precancerous Conditions".

4.3 Classification of drug-disease relations

4.3.1 Applicability of simple word analogy to drug-disease relations

Though word analogy is quite attractive, it does not always works well. To evaluate the applicability of word analogy to the discovery of new relation between drug and disease, checked whether most of the displacement vectors between confirmed drug-disease pairs (i.e. correct relations) are similar in length and parallel to each other or not. Unfortunately, as shown in Figure 4.4, the displacement vectors have wide range of lengths and directions. It indicates that the simple application of word analogy to drug repositioning cannot achieve high performance.

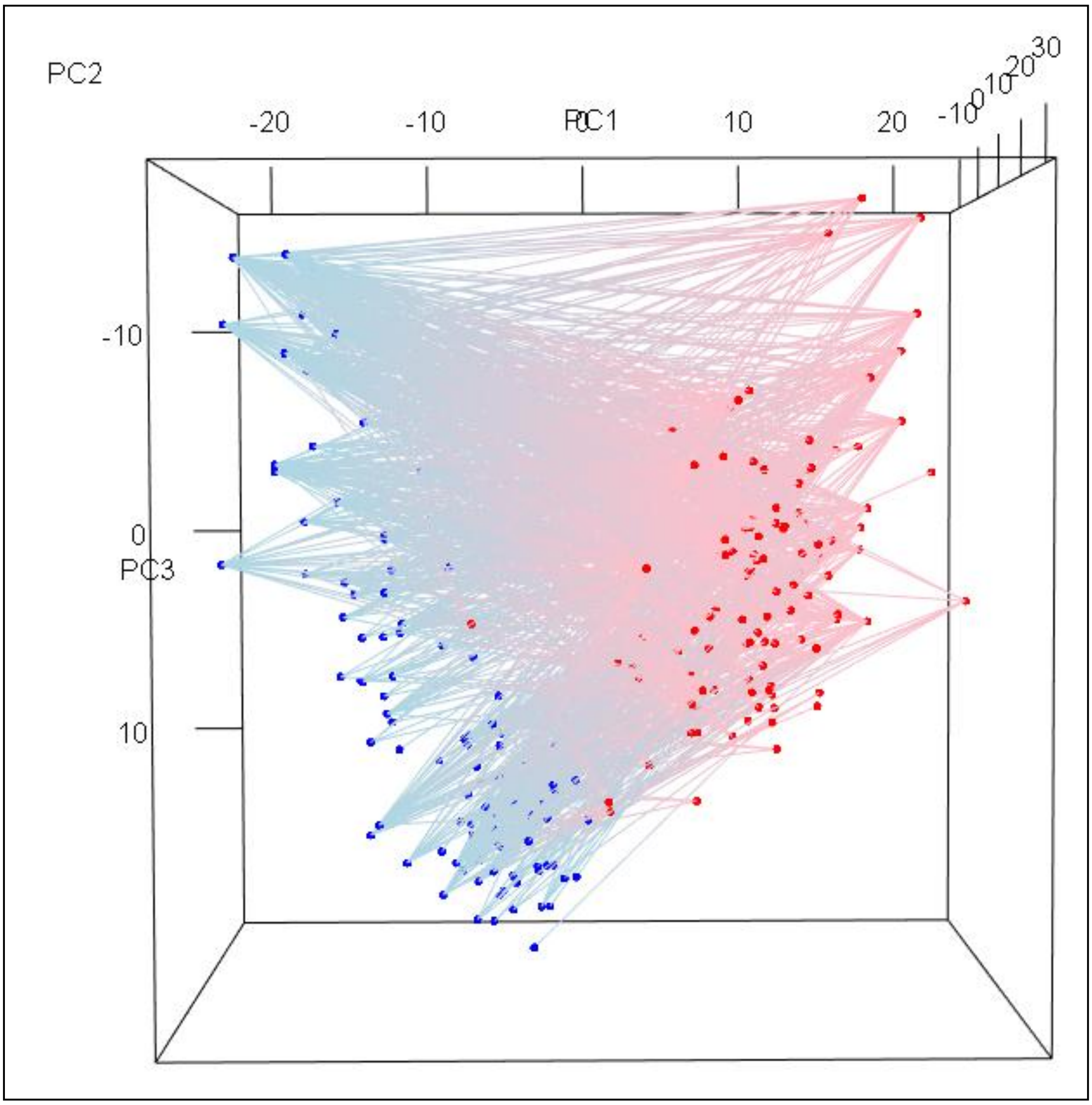


Figure 4.4 Distribution of displacement vectors for cancer-related drug-disease relations in CTD. Blue and red points represents anti-cancer drugs and cancer-related diseases.

4.3.2 Classification of unseen drug-disease relations

Instead of simple application of word analogy, we constructed a classification model using SVM. For all combinations of 104 anti-cancer drugs and 107 cancer-related diseases (i.e. 11,128 drug-disease pairs), drug vectors and disease vectors were concatenated and binary class labels (i.e. positive or negative) were added according to 1,097 correct drug-disease relations extracted from CTD. Due to the imbalance of two classes, 1,097 out of 10,031 negative examples were randomly selected so that the numbers of positive and negative examples are balanced (see Figure 3.6).

The result of performance evaluation is shown in Table 4.1. Each accuracy is an average of 100 times 10-fold cross-validation with different subsets of negative examples. In this table, it was revealed that the performance of classification was not so affected by vector size and window size, and the best accuracy was 87.6%. For exploratory use of the classification model to discover candidate drug-disease pairs, it means sufficiently high performance.

Table 4.1 Accuracy of classifying correct and incorrect drug-disease relations by SVM

vector size	**accuracy**			
	window size = 2	**window size = 3**	**window size = 4**	**window size = 8**
50	0.872	0.873	0.875	0.872
75	0.873	0.874	0.874	0.874
100	0.874	0.874	0.874	**0.876**
200	0.874	0.874	0.874	0.874

Finally, we tested all combinations of 2,199 drugs not used in training and 107 cancer-related diseases (in total, 235,293 drug-disease pairs). In case of the classification model trained by 11,128 examples, only 64 test examples were predicted as positive, and all the drugs in the examples were anti-cancer drugs (but not included in 104 anti-cancer drugs used for training). By controlling the degree of class imbalance in training data, it is possible to predict a pair of non-anti-cancer drug and cancer-related disease as positive. For example, using the classification model trained by

1,097 positive and 8,776 negative examples (degree of imbalance is 1:8), 10 times training and test by 235,293 drug-disease pairs discovered the following candidate drugs for repositioning to cancer treatment, where the numbers indicate how many times they were discovered in 10 times training and test.

drug::urokinase(10), drug::photodynamic_therapy(10), drug::oxygen(10), drug::nonoxynol-9(10), drug::nitroglycerin(10), drug::nitrogen(10), drug::l-phenylalanine(10), drug::l-methionine(10), drug::l-glutamine(10), drug::l-cysteine(10), drug::glutathione(10), drug::glucose(10), drug::epoxide(10), drug::enzyme(10), drug::collagenase(10), drug::bisphosphonate(10), drug::amino_acid(10), drug::amide(10), drug::clarithromycin(9), drug::vitamin(8), drug::l-proline(7), drug::vitamin_e(6), drug::xanthophyll(4), drug::phospholipid(4), drug::palifermin(4), drug::ether(4), drug::ethacrynic_acid(4), drug::denosumab(4), drug::egfr_inhibitor(2), drug::pyruvic_acid(1).

Besides too general names like “drug::enzyme” and “drug::amide”, it is notable that the above list includes approved anti-cancer drugs (e.g. “drug::denosumab”), anti-cancer drugs under investigation (e.g. “drug::clarithromycin”, “drug::bisphosphonate”, and “drug::xanthophyll”), and drugs potentially promote cancer (e.g. “drug::urokinase” and “drug::collagenase”). Especially, it should be emphasized that repositioning of clarithromycin to anti-cancer agent has been reported in 2015 [38], despite the fact that the corpus was downloaded in 2013. Though further screening based on expert’s knowledge is necessary, this result demonstrate that the classification of concatenated word vector is a promising approach to in-silico screening of drug-disease relations for drug repositioning.

Chapter 5

Conclusion and Future Works

In this chapter, we would like to summarize main works that we have done in our research. In order to go on this study in the future, we introduce some strategies for improving the result as well as expanding the scope of the research.

5.1 Dissertation summary

It can be said that the development of new and effective drugs always plays an important role for pharmaceutical companies and institutes. As a key technology of rapid and low-cost drug development, drug repositioning is getting more popular. In general, drug repositioning is reuse of existing drugs for other purposes.

Besides biological trial and error, computational approaches are actively tested for drug repositioning. In this study, we applied text mining techniques to drug repositioning in order to (i) check the distribution of biomedical words; (ii) analyze clustering of anti-cancer drug vectors and cancer-related disease vectors; (iii) find unseen drug-disease relations by classification.

One of the reasons why word embedding by word2vec becomes popular is its functionality of word analogy [6]. For example, if a sufficient amount of corpus is converted into word vectors and used in the analogy, it could predict fourth word “California” from three given words “Chicago”, “Illinois”, and “Stockton”. Since a state for a city is unique, it works well: it readily means that the analogy easily fails for one-to-many relationship (e.g. predicting “Stockton” from “Illinois”, “Chicago”, and “California”). About drug-disease relationship, at first we expected that one drug is used for basically one disease. However, as shown in Figure 4.4, it was one-to-many from both sides of drug-disease relation. For another problem like gene-protein relationship, accuracy of word analogy might be high since only one protein is produced from one gene, ignoring alternative splicing.

Although word analogy was not available, word2vec provided significant advantages in the text mining from large amount of biomedical texts in this study. It efficiently encoded more than 1.7 million words into quite short vectors (e.g. 200 dimensions). If we use traditional word frequency and vector space model, one vector for a word is a vector of 1.7 million features with extremely high sparsity. Due to the efficiency of encoding, we could process the whole corpus in reasonable memory space and computation time. Furthermore, the word vectors generated by word2vec seem well reflect the semantic space of biomedical words. Figure 4.1 illustrates that the words in different semantic categories are well separated in case of sufficiently high frequency of occurrences. Also, the results of clustering shown in Figures 4.2

and 4.3 indicate that similarities among words in the same category are also fine. They might be promising results for further application of word embedding in biomedical text mining.

In this study, it was revealed that word embedding is effective for representing sense of all words in large amount of cancer-related PubMed abstracts. Furthermore, concatenation of word vectors of drugs and diseases well represents their relations and could be used for finding candidate drugs for repositioning by classification.

5.2 Future works

In this study, we mainly focused on the prediction of drug-disease relations in order to find new indications of existing drugs for the treatment of cancer diseases. We hope that this model will also be applied to other kind of diseases in the future. Moreover, new kinds of biomedical relations such as gene-disease relations or drug-gene relations will be discovered by a similar model.

For better performance of classification, various feature selection and oversampling algorithms [39] will be tested in the future work.

To check the effects of quantity and quality of the training data on the quality of word vectors, we will attempt to cover whole PubMed abstracts.

Bibliography

[1] Ferreira, L.G., dos Santos, R.N., Oliva, G. and Andricopulo, A.D. (2015) Molecular docking and structure-based drug design strategies. *Molecules*, 20:13384–13421.

[2] Bajorath, J. (2015) Computer-aided drug discovery, *F1000Research 2015*, 4(F1000 Faculty Rev):630.

[3] Ashburn, T.T. and Thor, K.B. (2004) Drug repositioning: Identifying and developing new uses for existing drugs. *Nature Reviews Drug Discovery*, 3(8):673–683.

[4] Emig, D., Ivliev, A., Pustovalova, O., Lancashire, L., Bureeva, S., Nikolsky, Y., Bessarabova, M. (2013) Drug Target Prediction and Repositioning Using an Integrated Network-Based Approach. *PLoS ONE,* 8(4):e60618.

[5] Fellbaum, C. and Miller, G. (1998). *WordNet: An Electronic Lexical Database*. A Bradford Book.

[6] Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013) Efficient Estimation of Word Representations in Vector Space. in *Proceedings of Workshop at ICLR,* arXiv:1301.3781v1.

[7] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. and Dean, J. (2013) Distributed Representations of Words and Phrases and their Compositionality. in *Proceedings of NIPS,* arXiv:1301.3781v3.

[8] Mikolov, T., Yih, W.T. and Zweig. G. (2013) Linguistic Regularities in Continuous Space Word Representations. in *Proceedings of NAACL HLT*, 746-751.

[9] Castellano, M., Mastronardi, G., Aprile, A. and Tarricone, G. (2007) A web text mining flexible architecture. *International Journal of Computer Science and Engineering*, 1(4):252–259.

[10] Hotho, A., Nürnberger, A. and Paaß, G. (2005) A Brief Survey of Text Mining. *Ldv Forum*, 20(1), 19-62.

[11] Porter, M.F. (1980) An algorithm for suffix stripping. *Program*, 14(3):130-137.

[12] Hiemstra, D. (2001) *Using language models for information retrieval*. PhD dissertation, University of Twente.

[13] Salton, G., Wong, A. and Yang, C.S. (1975) A vector space model for automatic indexing. *Communications of the ACM,* 18(11):613-620.

[14] Salton, G. (1989) *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley.

[15] Vapnik, V. (1999) *The Nature of Statistical Learning Theory*, Second Edition, Springer.

[16] Joachims, T. (1998) Text categorization with support vector machines: Learning with many relevant features. in *Proceedings of European Conference on Machine Learning (ECML),* Berlin, Springer, 137-142.

[17] Berkhin, P. (2006) A survey of clustering data mining techniques. in *Grouping multidimensional data*, Springer Berlin Heidelberg, 25-71.

[18] Kaufman, L. and Rousseeuw, P.PJ. (2009) *Finding groups in data: An introduction to cluster analysis*. Wiley-Interscience.

[19] Guha, S., Rastogi, R. and Shim, K. (1998) CURE: An Efficient Clustering Algorithm for Large Databases. in *ACM SIGMOD Record*, 27(2):73-84.

[20] Tuason, O., Chen, L., Liu, H., Blake, J.A. and Friedman, C. (2004) Biological nomenclatures: a source of lexical knowledge and ambiguity. in *Proceedings of Pacific Symposium on Biocomputing 2004*, 238-249.

[21] Hirschman, L., Morgan, A.A. and Yeh, A.S. (2002) Rutabaga by any other name: extracting biological names. *Journal of Biomedical Informatics*, 35(4):247-259.

[22] Schuler, G.D., Epstein, J.A., Ohkawa, H. and Kans, J.A. (1996) Entrez: molecular biology database and retrieval system. *Methods in Enzymology*, 266:141-162.

[23] Website of PubMed [http://www.ncbi.nlm.nih.gov/pubmed].

[24] Website of MeSH [http://www.ncbi.nlm.nih.gov/mesh].

[25] The MeSH download page [http://www.ncbi.nlm.nih.gov/mesh/filelist.html].

[26] Whirl-Carrillo, M., McDonagh, E.M., Hebert, J.M., Gong, L., Sangkuhl, K., Thorn, C.F., Altman, R.B. and Klein, T.E. (2012) Pharmacogenomics Knowledge for Personalized Medicine. *Clinical Pharmacology & Therapeutics* 92(4):414-417.

[27] Website of PharmGKB [http://www.pharmGKB.org].

[28] Wishart, D.S., Knox, C., Guo, A.C., Shrivastava, S., Hassanali, M., Stothard, P., Chang, Z. and Woolsey, J. (2006) DrugBank: a comprehensive resource for in

silico drug discovery and exploration. *Nucleic Acids Res.*, 34(Database issue):D668-672.

[29] Website of DrugBank [http://www.drugbank.ca].

[30] Davis, A.P., Grondin, C.J., Lennon-Hopkins, K., Saraceni-Richards, C., Sciaky, D., King, B.L., Wiegers, T.C. and Mattingly, C.J. (2015) The Comparative Toxicogenomics Database’s 10th year anniversary: update 2015. *Nucleic Acids Res.*, 43(D1):D914-D920.

[31] Website of CTD [http://ctdbase.org/download].

[32] Website of Enju [http://www.nactem.ac.uk/enju]

[33] Website of word2vec [https://code.google.com/p/word2vec]

[34] Miyao, Y. and Tsujii, J. (2008) Feature forest models for probabilistic HPSG parsing. *Computational Linguistics*, 34(1):35–80.

[35] WHO Collaborating Centre for Drug Statistics Methodology, ATC classification index with DDDs, 2015. Oslo 2014.

[36] Lipscomb, C.E. (2000) Medical Subject Headings (MeSH), *Bulletin of the Medical Library Association*, 88 (3):265.

[37] Xu, R. and Wunsch, D.I.I. (2005) Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645-678.

[38] Pantziarka, P., Bouche, G., Meheus, L., Sukhatme, V. and Sukhatme, V.P. (2015) Repurposing Drugs in Oncology (ReDO)-clarithromycin as an anti-cancer agent. *ecancermedicalscience* 9:513.

[39] Dang, X.T., Hirose, O., Bui, D.H., Saethang, T., Tran, V.A., Nguyen, T.L.A., Le, T.T.K., Kubo, M., Yamada,Y. and Satou,K. (2013) A Novel Over-Sampling Method and its Application to Cancer Classification from Gene Expression Data. *Chem-Bio Informatics Journal*, 13:19-29.